



THESIS - SS142501

***DEEP LEARNING NEURAL NETWORK DAN  
GENETIC ALGORITHM PADA ANALISIS KLASIFIKASI  
STATUS PENERIMAAN BEASISWA BIDIKMISI***

(Studi Kasus : Beasiswa Bidikmisi Di Jawa Timur  
Tahun 2017)

NITA CAHYANI  
NRP. 06211650010012

DOSEN PEMBIMBING :  
Dr. Kartika Fithriasari, M.Si.  
Irhamah, M.Si., Ph.D.

PROGRAM MAGISTER  
DEPARTEMEN STATISTIKA  
FAKULTAS MATEMATIKA, KOMPUTASI, DAN SAINS DATA  
INSTITUT TEKNOLOGI SEPULUH NOPEMBER  
SURABAYA  
2018



THESIS - SS142501

**DEEP LEARNING NEURAL NETWORK AND GENETIC  
ALGORITHM CLASSIFICATION ANALYSIS STATUS  
ACCEPTANCE OF SCHOLARSHIP BIDIKMISI**

(Case Study : Bidikmisi Scholarship In East Java  
Year 2017)

NITA CAHYANI  
NRP. 06211650010012

SUPERVISOR :  
Dr. Kartika Fithriasari, M.Si.  
Irhamah, M.Si., Ph.D.

MAGISTER PROGRAMME  
DEPARTEMENT OF STATISTICS  
FACULTY OF MATHEMATICS, COMPUTING, AND DATA SCIENCE  
INSTITUT TEKNOLOGI SEPULUH NOPEMBER  
SURABAYA  
2018

**DEEP LEARNING NEURAL NETWORK DAN GENETIC ALGORITHM  
PADA ANALISIS KLASIFIKASI STATUS PENERIMAAN BEASISWA  
BIDIKMISI**


**(Studi Kasus : Beasiswa Bidikmisi Di Jawa Timur Tahun 2017)**

Tesis disusun untuk memenuhi salah satu syarat memperoleh gelar  
Magister Sains (M.Si)  
di  
Institut Teknologi Sepuluh Nopember

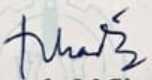
Oleh :  
**NITA CAHYANI**  
**NRP. 06211650010012**

Tanggal Ujian : 10 Juli 2018  
Periode Wisuda : September 2018

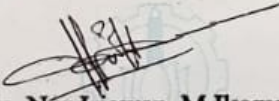
Disetujui Oleh :

1.   
Dr. Kartika Fithriasari, M.Si.  
NIP. 19691212 199303 2 002

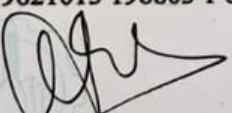
(Pembimbing I)

2.   
Irhamah, M.Si., Ph.D.  
NIP. 19780406 200112 2 002

(Pembimbing II)

3.   
Prof. Drs. Nur Iriawan, M.Ikomp., Ph.D.  
NIP. 19621015 198803 1 002

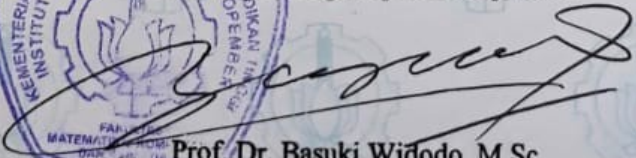
(Penguji)

4.   
Dr. Agnes Tuti Rumiati, M.Sc.  
NIP. 19570724 198503 2 002

(Penguji)

Dekan  
Fakultas Matematika, Komputasi, dan Sains Data  
Institut Teknologi Sepuluh Nopember



  
Prof. Dr. Basuki Widodo, M.Sc.  
NIP. 19650605 198903 1 002

# ***DEEP LEARNING NEURAL NETWORK DAN GENETIC ALGORITHM PADA ANALISIS KLASIFIKASI STATUS PENERIMAAN BEASISWA BIDIKMISI***

(Studi Kasus : Beasiswa Bidikmisi Di Jawa Timur Tahun 2017)

Nama Mahasiswa : Nita Cahyani  
NRP : 06211650010012  
Pembimbing : Dr. Kartika Fithriasari, M.Si.  
Co-Pembimbing : Irhamah, M.Si., Ph.D.

## **ABSTRAK**

*Neural Network* merupakan salah satu alat analisis data mining yang dapat digunakan untuk membuat klasifikasi pada data pendaftaran beasiswa Bidikmisi di Jawa Timur. Pemilihan model pada *Neural Network* memerlukan berbagai faktor seperti pemilihan jumlah *hidden node* yang optimal, pemilihan variabel *input* yang relevan dan pemilihan bobot koneksi yang optimal. *Genetic Algorithm* (GA) merupakan suatu teknik optimasi yang didasarkan pada prinsip genetika dan seleksi alam. Dalam *Genetic Algorithm* populasi terbentuk dari banyak individu yang berkembang sesuai aturan seleksi spesifik dengan memaksimalkan fitness. *Genetic Algorithm* digunakan untuk mendapatkan nilai global optimum dengan cara melakukan perulangan atau iterasi pada konsep evolusi Darwin. Penelitian ini menyajikan metode *Deep learning Neural Network* dengan *Backpropagation Network* (BPNN) dengan dua *hidden layer* dan metode gabungan BPNN dengan GA, dimana GA digunakan untuk menginisialisasi dan mengoptimasi bobot koneksi BPNN. Hasil analisis menunjukkan bahwa metode optimasi bobot koneksi BPNN dengan GA memberikan nilai ketepatan klasifikasi yang lebih baik dibandingkan dengan metode BPNN tanpa optimasi bobot.

**Kata kunci:** *Backpropagation Neural Network* (BPNN), Bidikmisi, *Deep Learning Neural Network*, *Genetic Algorithm* (GA), *Neural Network* (NN).

*(halaman ini sengaja dikosongkan)*

**DEEP LEARNING NEURAL NETWORK AND GENETIC  
ALGORITHM IN CLASSIFICATION ANALYSIS STATUS  
ACCEPTANCE OF SCHOLARSHIP BIDIKMISI  
Case Study: Bidikmisi Scholarship In East Java Year 2017**

Name	: Nita Cahyani
Student Identity Number	: 06211650010012
Supervisor	: Dr. Kartika Fithriasari, M.Si
Co-Supervisor	: Irhamah, M.Si, Ph.D

**ABSTRACT**

Neural Network is one of the data mining analysis tools that can be used to create data on registration data of Bidikmisi students in East Java. Model selection on neural networks. Many factors such as the selection of the optimal number of hidden nodes, the relevant input selection variables and the optimal connection speed. Genetic Algorithm (GA) is a technique used on genetic and natural principles. Genetic Algorithms are formed by many Developed individuals according to the rules. Genetic Algorithm to obtain optimal global value by means of looping or iterating on the concept of the Darwinian revolution. This research uses Deep Learning Neural Network with Backpropagation Network (BPNN) with two hidden layers and BPNN combined method with GA, where GA to initialize and optimize the weight of BPNN connection. The results showed that the BPNN connection optimization method with GA gives better accuracy value compared to the BPNN method without weight optimization.

**Keywords:** Backpropagation Network (BPPN), Bidikmisi, Deep Learning Neural Network, Genetic Algorithm (GA), Neural Network (NN).

*(halaman ini sengaja dikosongkan)*

## KATA PENGANTAR

Segala puji dan syukur bagi Allah SWT yang telah memberikan rahmat, hidayah, dan kasih sayang-Nya, sehingga penulis dapat menyelesaikan Tesis dengan judul “*Deep Learning Neural Network Dan Genetic Algorithm Pada Analisis Klasifikasi Status Penerimaan Beasiswa Bidikmisi (Studi Kasus: Beasiswa Bidikmisi di Jawa Timur Tahun 2017)*”.

Selesainya Tesis ini tidak lepas dari bantuan berbagai pihak, untuk itu penulis ingin menyampaikan rasa terima kasih yang sebesar-besarnya kepada :

1. Keluargaku tercinta Suamiku Febrianto, Ibunda Mahdianah, Ayahanda Edriyan, Ibu Mertua, Bapak Mertua, Kakak Amin, Adik Nur, Adik Munady dan seluruh keluarga yang telah memberikan semangat dan doa kepada penulis dalam menyelesaikan studi ini.
2. Pembimbing penulis Ibu Dr. Kartika Fithriasari, M.Si. dan Ibu Irhamah, M.Si., Ph.D. yang telah banyak meluangkan waktunya untuk memberikan petunjuk, bimbingan dan arahan kepada penulis dalam menyelesaikan Tesis ini.
3. Bapak Prof. Drs. Nur Iriawan, M.Ikomp., Ph.D. dan Ibu Dr. Agnes Tuti Rumiati, M.Sc. selaku dosen penguji yang telah memberikan banyak saran dan masukan agar Tesis ini menjadi lebih baik.
4. Bapak Dr. Suhartono, M.Sc. selaku Ketua Departemen Statistika ITS dan Bapak Dr. rer. Pol. Heri Kuswanto, M.Si. selaku Kaprodi Pascasarjana Statistika FMKSD ITS.
5. Ibu Dr. Ismaini Zain, M.Si. selaku dosen wali yang telah memberikan arahan dan masukan kepada penulis selama berada di lingkungan kampus ITS Surabaya.
6. Bapak, Ibu dosen dan staf Departemen Statistika FMKSD ITS yang dengan tulus ikhlas telah memberikan ilmu dan bantuan selama penulis menyelesaikan studi.
7. Ibu Yuni atas segala bantuan dan dukungannya dan semua teman-teman seperjuangan mahasiswa S2 Statistika angkatan 2016/2017 Neni, Laila, Sinta, Farida, Indah, Febri, Silvi, Dwi, Alfian, Mila, Zakya, Sella, Dafiah atas segala



bantuan, kekompakan dan kebersamaan selama penulis berada di lingkungan kampus ITS Surabaya.

8. Semua pihak yang telah banyak membantu penulis, namun tidak dapat penulis sebutkan satu per satu.

Penulis mengharapkan kritik dan saran yang bersifat membangun demi kesempurnaan Tesis ini. Penulis berharap Tesis ini dapat memberikan manfaat dan sumbangan untuk menambah wawasan keilmuan bagi pembacanya.

Surabaya, Juli 2018

Penulis

## DAFTAR ISI

	Halaman
<b>HALAMAN JUDUL</b> .....	i
<b>LEMBAR PENGESAHAN</b> .....	v
<b>ABSTRAK</b> .....	vii
<b>ABSTRACT</b> .....	ix
<b>KATA PENGANTAR</b> .....	xi
<b>DAFTAR ISI</b> .....	xiii
<b>DAFTAR TABEL</b> .....	xv
<b>DAFTAR GAMBAR</b> .....	xix
<b>DAFTAR LAMPIRAN</b> .....	xxi
<b>BAB 1 PENDAHULUAN</b>	
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	5
1.3 Tujuan Penelitian .....	5
1.4 Manfaat Penelitian .....	6
1.5 Batasan Masalah .....	6
<b>BAB 2 TINJAUAN PUSTAKA</b>	
2.1 Data Mining .....	7
2.2 Jaringan Syaraf Tiruan .....	8
2.2.1 <i>Backpropagation</i> .....	11
2.3 Algoritma Genetika .....	14
2.3.1 Pengkodean .....	16
2.3.2 <i>Fitness</i> .....	16
2.3.3 Seleksi Orang Tua .....	16
2.3.4 Pindah Silang .....	17
2.3.5 Mutasi .....	17
2.3.6 Elitisme .....	17
2.3.7 Klasifikasi .....	17
2.4 Evaluasi Performasi Metode Klasifikasi .....	18
2.5 <i>Cross Validation</i> .....	19

2.6 Bidikmisi .....	20
2.6.1 Misi .....	21
2.6.2 Tujuan.....	21
2.6.3 Sasaran .....	21
2.6.4 Persyaratan dan Kuota .....	22
2.6.5 Mekanisme Seleksi .....	23
<b>BAB 3 METODOLOGI PENELITIAN</b>	
3.1 Sumber Data dan Variabel Penelitian.....	31
3.2 Struktur Data .....	34
3.3 Prosedur Penelitian .....	35
<b>BAB 4 ANALISIS DATA DAN PEMBAHASAN</b>	
4.1 Alur Proses Data .....	41
4.2 Analisis Karakteristik Data .....	44
4.3 Prediksi <i>Neural Network Backpropagation</i> .....	53
4.3.1 Model Prediksi <i>Neural Network 1 Hidden Layer</i> .....	54
4.3.2 Model Prediksi <i>Neural Network 2 Hidden Layer</i> .....	62
4.4 Optimasi Parameter <i>Neural Network Backpropagation</i> Menggunakan Algoritma Genetika (GA) .....	70
4.5 Tingkat Kinerja Klasifikasi <i>Neural Network Backpropagation</i> Tanpadaan Dengan Optimasi Bobot dan Bias <i>Backpropagation</i> .....	75
<b>BAB 5 KESIMPULAN DAN SARAN</b>	
5.1 Kesimpulan .....	77
5.2 Saran .....	78
<b>DAFTAR PUSTAKA .....</b>	<b>79</b>
<b>LAMPIRAN .....</b>	<b>83</b>
<b>BIOGRAFI PENULIS .....</b>	<b>103</b>

## DAFTAR TABEL

	Halaman
<b>Tabel 2.1</b> Pemetaan proses alamiah ke dalam proses GA .....	15
<b>Tabel 2.2</b> Confusion Matriks .....	18
<b>Tabel 3.1</b> Deskripsi Variabel Penelitian.....	31
<b>Tabel 3.2</b> Struktur Data .....	34
<b>Tabel 4.1</b> Proporsi Status Penerimaan Beasiswa Bidikmisi .....	42
<b>Tabel 4.2</b> Karakteristik Berdasarkan Status Penerimaan Beasiswa Bidikmisi dan Jenis Pekerjaan Ayah.....	44
<b>Tabel 4.3</b> Karakteristik Berdasarkan Status Penerimaan Beasiswa Bidikmisi dan Jenis Pekerjaan Ibu .....	45
<b>Tabel 4.4</b> Karakteristik Berdasarkan Status Penerimaan Beasiswa Bidikmisi dan Pendidikan Ayah .....	46
<b>Tabel 4.5</b> Karakteristik Berdasarkan Status Penerimaan Beasiswa Bidikmisi dan Pendidikan Ibu.....	47
<b>Tabel 4.6</b> Karakteristik Berdasarkan Status Penerimaan Beasiswa Bidikmisi dan Penghasilan Ayah .....	47
<b>Tabel 4.7</b> Karakteristik Berdasarkan Status Penerimaan Beasiswa Bidikmisi dan Penghasilan Ibu .....	48
<b>Tabel 4.8</b> Karakteristik Berdasarkan Status Penerimaan Beasiswa Bidikmisi dan Kepemilikan Rumah .....	49
<b>Tabel 4.9</b> Karakteristik Berdasarkan Status Penerimaan Beasiswa Bidikmisi dan Sumber Listrik.....	50
<b>Tabel 4.10</b> Karakteristik Berdasarkan Status Penerimaan Beasiswa Bidikmisi dan Luas Tanah .....	50
<b>Tabel 4.11</b> Karakteristik Berdasarkan Status Penerimaan Beasiswa Bidikmisi dan Luas Bangunan.....	51
<b>Tabel 4.12</b> Karakteristik Berdasarkan Status Penerimaan Beasiswa Bidikmisi dan Kepemilikan MCK .....	52

<b>Tabel 4.13</b> Karakteristik Berdasarkan Status Penerimaan Beasiswa Bidikmisi dan Jumlah Tanggungan.....	53
<b>Tabel 4.14</b> Tingkat Kinerja AUC, G-Mean, dan Akurasi Data <i>Testing</i> dan Data <i>Training</i> dengan 1 <i>Hidden Layer</i> .....	54
<b>Tabel 4.15</b> Ketepatan Klasifikasi 10-Fold <i>Cross Validation Neural Network Backpropagation</i> pada data <i>Training</i> dengan 1 <i>hidden Layer</i> .....	55
<b>Tabel 4.16</b> Rata-rata Ketepatan Klasifikasi Perkatagori Untuk 10-Fold Data <i>Training</i> Pada Model <i>Neural Network Backpropagation (1 Hidden Layer)</i> .....	57
<b>Tabel 4.17</b> Ketepatan Klasifikasi 10-Fold <i>Cross Validation Neural Network Backpropagation</i> pada data <i>Testing</i> dengan 1 <i>Hidden Layer</i> .....	59
<b>Tabel 4.18</b> Rata-rata Ketepatan Klasifikasi Perkatagori Untuk 10-Fold Data <i>Testing</i> Pada Model <i>Neural Network Backpropagation (1 Hidden Layer)</i> .....	60
<b>Tabel 4.19</b> Hasil kinerja klasifikasi dengan jumlah neuron 4 pada 1 <i>hidden layer</i> .....	61
<b>Tabel 4.20</b> Tingkat Kinerja AUC, G-Mean, dan Akurasi Data <i>Testing</i> dan Data <i>Training</i> dengan 2 <i>Hidden Layer</i> .....	62
<b>Tabel 4.21</b> Ketepatan Klasifikasi dengan 2 <i>Hidden layer 10-Fold Cross Validation Neural Network Backpropagation</i> pada data <i>Training</i> ...	63
<b>Tabel 4.22</b> Rata-rata Ketepatan Klasifikasi Perkatagori Untuk 10-Fold Data <i>Training</i> Pada Model <i>Neural Network Backpropagation (2 Hidden Layer)</i> .....	65
<b>Tabel 4.23</b> Ketepatan Klasifikasi dengan 2 <i>Hidden layer 10-Fold Cross Validation Neural Network Backpropagation</i> pada data <i>Testing</i> .....	66
<b>Tabel 4.24</b> Rata-rata Ketepatan Klasifikasi Perkatagori Untuk 10-Fold Data <i>Testing</i> Pada Model <i>Neural Network Backpropagation (2 Hidden Layer)</i> .....	68
<b>Tabel 4.25</b> Hasil Kinerja Klasifikasi 10-Fold Dengan Jumlah Neuron (10-4) Dengan 2 <i>Hidden Layer</i> Pada Data <i>Training</i> Dan <i>Testing</i> .....	70

<b>Tabel 4.26</b> Hasil Kinerja Klasifikasi BPN-GA dengan 4 neuron pada 1 <i>Hidden Layer</i> .....	74
<b>Tabel 4.27</b> Kinerja Hasil Klasifikasi Tanpa dan dengan Optimasi Bobot dan Bias Pada <i>Neural Network Backpropagation</i> .....	75

*(halaman ini sengaja dikosongkan)*

## DAFTAR GAMBAR

	Halaman
<b>Gambar 2.1</b> Model Neuron Biologi.....	8
<b>Gambar 2.2</b> Arsitektur Jaringan Syaraf Tiruan Multi Layer.....	10
<b>Gambar 2.3</b> Arsitektur <i>Backpropagation</i> .....	12
<b>Gambar 2.4</b> Proses SeleksiPenerimaan Beasiswa Bidikmisi.....	29
<b>Gambar 3.1</b> Arsitektur JST .....	37
<b>Gambar 3.2</b> Pengkodean Kromosom.....	37
<b>Gambar 3.3</b> Diagram Alir Keseluruhan Penelitian .....	38
<b>Gambar 3.4</b> Diagram Alir <i>Backpropagation</i> .....	39
<b>Gambar 3.5</b> Diagram Alir Optimasi Bobot BPN dengan GA.....	40
<b>Gambar 4.1</b> Proses Seleksi Pada Jumlah ID Tahapan Bidikmisi.....	41
<b>Gambar 4.2</b> Sebaran Data Status Beasiswa Bidikmisi .....	43
<b>Gambar 4.3</b> Sebaran Hasil Klasifikasi Data <i>Training 1 Hidden Layer</i> .....	57
<b>Gambar 4.4</b> Sebaran Hasil Klasifikasi Data <i>Testing 1 Hidden Layer</i> .....	60
<b>Gambar 4.6</b> Sebaran Hasil Klasifikasi Data <i>Training 2 Hidden Layer</i> .....	66
<b>Gambar 4.5</b> Sebaran Hasil Klasifikasi Data <i>Testing 2 Hidden Layer</i> .....	69
<b>Gambar 4.6</b> Pembentukan Kromosom Algoritma Genetika .....	72



*(halaman ini sengaja dikosongkan)*

## DAFTAR LAMPIRAN

	Halaman
<b>Lampiran A.</b> Data Penelitian Bidikmisi Provinsi Jawa Timur Tahun 2017 .....	83
<b>Lampiran B.</b> Syntax Membagi Data Menjadi 10-Fold Cross Validation .....	84
<b>Lampiran C.</b> Syntax <i>Neural Network</i> 1 Hiden Layer 10-Fold Cross Validation .....	90
<b>Lampiran D.</b> Syntax Optimasi Bobot <i>Backpropagation</i> Dengan 1 <i>Hidden Layer</i> Dengan 4 Neuron .....	92
<b>Lampiran E.</b> Jaringan Syaraf Tiruan <i>Backpropagation</i> (Contoh Manual Algoritma <i>Backpropagation</i> ).....	97

*(halaman ini sengaja dikosongkan)*

# **BAB 1**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Indonesia merupakan bangsa yang kaya akan sumber daya alamnya dan bangsa dengan status sebagai Negara ke empat dengan jumlah penduduk terbanyak di dunia, sehingga Indonesia seharusnya dapat memanfaatkan sumber daya alam yang ada untuk kemajuan bangsa. Tetapi dalam hal ini Indonesia belum memanfaatkan sumber daya yang ada, khususnya sumber daya manusia. Terlihat dari tingginya angka pengangguran di Indonesia, yang mencapai jutaan penduduk. Terjadinya pengangguran disebabkan oleh banyak faktor, salah satunya adalah pendidikan.

Pentingnya Pendidikan merupakan salah satu cara yang dapat memajukan suatu bangsa. Berdasarkan Pasal 31, Ayat (1) Undang-Undang Dasar 1945 bahwa, Tiap-tiap warga Negara berhak mendapatkan pengajaran, sehingga dalam hal ini pemerintah dan pemerintah daerah sudah sewajibnya memberikan layanan dan kemudahan serta menjamin terselenggaranya pendidikan yang bermutu bagi setiap warga negara tanpa diskriminasi, dan masyarakat berkewajiban memberikan dukungan sumber daya dalam penyelenggaraan pendidikan.

Biaya merupakan masalah persoalan dalam pendidikan, semakin tinggi pendidikan yang ditempuh, semakin besar biaya yang dibutuhkan. Hal ini terbukti dari permasalahan peningkatan pemerataan akses ke perguruan tinggi jenjang pendidikan menengah yang terdiri atas lulusan SMA/SMK/MA atau bentuk lain yang sederajat sampai saat ini masih merupakan masalah di negara kita. Banyak lulusan jenjang pendidikan menengah yang berprestasi dan merupakan calon mahasiswa yang potensial tidak dapat melanjutkan ke jenjang pendidikan tinggi karena berasal dari keluarga kurang mampu. Selain itu peningkatan akses informasi terhadap sumber pendanaan masih sangat terbatas. Upaya yang dapat dilakukan antara lain dengan menyusun data basesiswa jenjang pendidikan menengah yang cerdas dan kurang mampu serta memfasilitasi atau menyediakan beasiswa dan biaya pendidikan (Departemen Pendidikan Nasional, 2010).

Mengacu pada Undang-Undang dan Peraturan Pemerintah serta kenyataan tentang program beasiswa sebagaimana tersebut di atas, maka Pemerintah melalui Direktorat Jenderal Pendidikan Tinggi Departemen Pendidikan Nasional mulai tahun 2010 memberikan beasiswa dan biaya pendidikan bagi calon mahasiswa dari keluarga yang kurang mampu secara ekonomi dan berprestasi yang disebut Beasiswa Bidikmisi (Departemen Pendidikan Nasional, 2010). Pemberian beasiswa merupakan program kerja yang ada di setiap Kabupaten/Kota di tiap-tiap perguruan tinggi, dalam hal ini khususnya Kabupaten Jawa Timur. Dalam penentuan beasiswa dilakukan dengan beberapa tahapan yaitu pendaftaran dan penyeleksian. Sehingga dibutuhkan model yang mampu memprediksi dengan akurasi tinggi dalam menentukan ketepatan dalam penerimaan beasiswa.

Klasifikasi merupakan metode statistika yang biasanya digunakan untuk menentukan ketepatan atau memetakan penerimaan pendaftar beasiswa Bidikmisi serta untuk mengetahui seberapa besar keakuratan dari hasil prediksi terhadap observasi yang sebenarnya. Terdapat beberapa metode klasifikasi yang biasanya digunakan yaitu metode klasik dan metode modern. Metode klasik yang sering digunakan yaitu metode Regresi Logistik dan Analisis Diskriminan sedangkan untuk metode modern yaitu *Neural Network*, *Genetica Algorithm* dan *Support Vector Machine*.

Oladokun & Adebajo (2008) melakukan penelitian tentang kualitas lulusan Universitas Ibadan di Nigeria. Dikatakan bahwa telah di selidiki dan terdapat kekurangan pada system pemeriksaan pendaftaran di Universitas Nasional. Digunakan metode *Artificial Neural Network* (ANN) model khususnya *Multi Layer Perceptron* untuk memprediksi kemungkinan kinerja calon siswa yang akan dipertimbangkan untuk masuk ke universitas tersebut, berbagai faktor yang diduga mempengaruhi kinerja siswa tersebut anatara lain nilai mata pelajaran, nilai ujian materikulasi, usia saat masuk Universitas, latar belakang orang tua, jenis dan lokasi sekolah menengah dan jenis kelamin. Hasil menunjukkan bahwa model ANN mampu memprediksi performa calon siswa dengan benar sebesar 70%.

Lye dkk (2010) melakukan penelitian dengan menggunakan tiga metode, yaitu *Backpropagation Neural Network* (BPNN), *Generalized Regression Neural*

Network (CART), dan Generalised Regression Neural Network (GRNN) dalam memprediksi prestasi matematika siswa. Bagian pertama dari penelitian ini menggunakan data pendaftaran untuk memprediksi hasil evaluasi semester siswa, sedangkan bagian terakhir menggunakan data tambahan untuk memprediksi hasil akhir penelitian siswa. Akurasi model prediktif dievaluasi dengan menggunakan crossvalidasi 10 kali lipat untuk mengidentifikasi model terbaik. Penelitian tersebut menghasilkan bahwa metode BPNN lebih baik dibanding menggunakan metode lain dengan akurasi 66,67% dan 71,11% .

Beberapa model statistik tentang klasifikasi yang dikembangkan dalam bidang pendidikan lainnya yaitu (Walczak & Sincich, 1999) meneliti tentang penerimaan mahasiswa di perguruan tinggi dengan membandingkan metode analisis regresi dengan *Neural Network*, kemudian Teshnizi, dkk (2015) meneliti dengan membandingkan regresi logistik dengan *Neural Network* untuk memprediksi kegagalan akademik siswa dengan menghasilkan masing-masing klasifikasi 77,5% untuk hasil klasifikasi metode regresi logistik dan 84,3% untuk klasifikasi metode *Neural Network*. Dihasilkan metode *Neural Network* lebih baik dibandingkan dengan metode regresi logistik.

Sayekti, 2013 meneliti tentang pengujian model jaringan syaraf tiruan untuk kualifikasi calon mahasiswa baru program Bidikmisi di Politeknik Negeri Semarang. Dengan menggunakan 8 variabel *input*, yang terdiri dari  $X_1$  (Pekerjaan Ayah),  $X_2$  (Pekerjaan Ibu),  $X_3$  (Penghasilan Ayah),  $X_4$  (Penghasilan Ibu),  $X_5$  (Jumlah Tanggungan),  $X_6$  (Pendidikan Ayah),  $X_7$  (Pendidikan Ibu) dan  $X_8$  (Nilai Akademik), diperoleh hasil bahwa sistem yang dibangun dengan jaringan syaraf tiruan *Backpropagation* mampu mengkualifikasi calon penerima beasiswa Bidik Misi dengan tingkat keberhasilan mencapai 99,21 % dengan menggunakan fungsi aktivasi sigmoid biner pada lapisan *input* dan fungsi identitas untuk lapisan tersembunyi terhadap lapisan *output*.

*Genetic Algorithm* (GA) merupakan suatu teknik optimasi yang didasarkan pada prinsip genetika dan seleksi alam. Dalam algoritma Genetika populasi terbentuk dari banyak individu yang berkembang sesuai aturan seleksi spesifik dengan memaksimalkan fitness (Haupt & Haupt, 2004). Algoritma ini juga digunakan untuk mendapatkan nilai global optimum dengan cara melakukan

perulangan atau iterasi pada konsep evolusi Darwin. Dalam teori evolusi Darwin, suatu individu tercipta secara acak kemudian berkembang biak melalui proses reproduksi sehingga terbentuk sekumpulan individu sebagai suatu populasi. Setiap individu dalam populasi mempunyai tingkat *fitness*/kebugaran yang berbeda-beda. Tingkat kebugaran ini menentukan seberapa kuat untuk tetap bertahan hidup dalam populasinya. Sebagian individu tetap bertahan hidup dan sebagian lainnya mati.

Menurut Gen & Cheng (1997) Algoritma Genetika mempunyai kelebihan yaitu melakukan sedikit perhitungan matematis yang berhubungan dengan masalah yang ingin diselesaikan. Karena sifat perubahan evolusi alamiahnya, sehingga algoritma ini akan mencari penyelesaian tanpa memperhatikan proses-proses yang berhubungan dengan masalah yang diselesaikan secara langsung. Goldberg (1989) juga mengatakan bahwa algoritma genetika mencari penyelesaian dari titik-titik populasi dan bukan dari sebuah titik saja. Proses pencarian dengan sekumpulan titik-titik populasi ini menyebabkan algoritma genetika lebih kecil kemungkinan untuk terjebak pada nilai optimum lokal.

Pembelajaran *Backpropagation Network* (BPN) bekerja dengan membuat modifikasi pada nilai bobot yaitu dari lapisan *output* kemudian bergerak mundur melalui jaringan lapisan tersembunyi. BPN menggunakan metode gradien untuk menemukan bobot dan cenderung menimbulkan masalah seperti masalah minimum lokal, kecepatan konvergensi yang lambat dan ketidakstabilan konvergensi dalam prosedur pelatihannya. Tidak seperti kebanyakan algoritma lainnya, yang hanya melakukan pencarian lokal, GA melakukan pencarian global. GA adalah prosedur iteratif yang terdiri atas populasi berukuran konstan dari individu yang disebut kromosom, masing-masing diwakili oleh string simbol yang terbatas, yang dikenal sebagai genom, yang mengkodekan solusi dalam masalah yang diberikan. GA dapat digunakan untuk memperbaiki kinerja BPN dengan cara yang berbeda (Karegowda, dkk 2011).

Karegowda, dkk (2011) mengatakan bahwa GA adalah metode pencarian umum stokastik, yang mampu secara efektif menjelajahi ruang pencarian yang besar, yang telah digunakan dengan BPN untuk menentukan jumlah node tersembunyi dan lapisan tersembunyi, pilih subset fitur yang relevan, tingkat

pembelajaran, momentum, dan inisialisasi dan optimalkan bobot koneksi jaringan BPN. Ahmed, dkk (2009) menggunakan GA untuk mendesain parameter ANN dengan optimal, termasuk arsitektur ANN, bobot, pemilihan *input*, fungsi aktivasi, tipe ANN, algoritma pelatihan, dan jumlah iterasi.

Berdasarkan uraian diatas penelitian ini akan menggunakan metode *Neural Network Backpropagation* yaitu dengan menggunakan dua *hidden layer* atau *Deep learning Neural Network* (DLNN) dan menggunakan GA pada BPN untuk mengoptimasi bobot pada pelatihan BPN yang tujuannya membandingkan kedua metode tersebut untuk mendapatkan tingkat kinerja yang lebih baik yang diharapkan hasil yang dicapai akan memberikan ketepatan klasifikasi dalam menentukan ketepatan sasaran penerimaan beasiswa Bidikmisi di Jawa Timur atau memberikan keputusan yang lebih tepat dalam pertimbangan penetapan mahasiswa yang diterima.

## **1.2 Rumusan Masalah**

Berdasarkan latar belakang yang telah dikembangkan diatas, maka rumusan masalah yang akan dibahas dalam penelitian ini yaitu bagaimana ketepatan klasifikasi pada penerimaan beasiswa Bidikmisi di Jawa Timur dengan menggunakan metode *Backpropagation Network* (BPN), *Deep Learning Neural Network* (DLNN) dan menggunakan metode *Genetic Algorithm* (GA) dalam mengoptimasi bobot di *Artificial Neural Network* (ANN) serta membandingkan ketiga metode tersebut yang diharapkan hasil yang dicapai akan memberikan ketepatan klasifikasi dalam menentukan ketepatan sasaran penerimaan beasiswa Bidikmisi di Jawa Timur.

## **1.3 Tujuan Penelitian**

Berdasarkan rumusan masalah diatas, tujuan yang ingin dicapai dalam penelitian ini adalah sebagai berikut.

1. Mengetahui ketepatan klasifikasi yang didapatkan dengan menggunakan metode *Backpropagation Network* (BPN) dan *Deep Learning Neural*



*Network* (DLNN) pada data siswa pendaftaran beasiswa Bidikmisi di Jawa Timur.

2. Mengetahui ketepatan klasifikasi yang didapatkan dengan menggunakan metode hybrid ANN-GA pada data siswa pendaftaran beasiswa Bidikmisi di Jawa Timur.
3. Mengetahui Perbandingan diantara BPN, DLNN dan ANN-GA.

#### **1.4 Manfaat Penelitian**

Manfaat yang ingin dicapai pada penelitian ini yaitu sebagai bahan pertimbangan dalam menentukan ketepatan penerimaan sasaran dalam pendaftaran siswa beasiswa Bidikmisi di Jawa Timur guna memberikan beasiswa Bidikmisi bagi orang-orang yang memang benar-benar berprestasi dan dari keluarga yang tidak mampu dan hasil yang didapatkan dapat dijadikan evaluasi bahwa ada ketidak tepatan alokasi dalam Bidikmisi yang di terima atau tidak diterima.

#### **1.5 Batas Penelitian**

1. Penelitian ini menggunakan metode *Backpropagation* dengan 1 *hidden layer* dan *Deep Learning Neural Network* (DLNN) yaitu *Backpropagation* dengan menggunakan 2 *hidden layer*.
2. Algoritma Genetika atau *Genetic Algorithm* (GA) digunakan dalam mengoptimasi bobot dan bias awal pada ANN (hybrid ANN-GA) dengan algoritma *Backpropagation* (1 *hidden layer*).
3. Data yang digunakan adalah data Bidikmisi di Jawa Timur Tahun 2017.

## BAB 2

### TINJAUAN PUSTAKA

Pada bab ini akan diuraikan teori singkat metode yang digunakan untuk memperoleh hasil klasifikasi data beasiswa Bidikmisi, yaitu Data Mining, Jaringan Syaraf Tiruan, Algoritma *Backpropagation*, Algoritma Genetika, dan Bidikmisi

#### 2.1 Data Mining

Data Mining adalah proses yang menggunakan teknik statistik, matematika, kecerdasan buatan, dan *machine learning* untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terkait dari berbagai database besar (Turban, dkk. 2005). Istilah data mining dan *knowledge discovery in databases* (KDD) seringkali digunakan secara bergantian untuk menjelaskan proses penggalian informasi tersembunyi dalam suatu basis data yang besar. Salah satu tahapan dalam keseluruhan proses KDD adalah *data mining* (Luthfi, dkk. 2009).

Proses KDD secara garis besar dapat dijelaskan sebagai berikut (Fayyad, 1996):

1. *Data Selection*

Pemilihan (seleksi) data dari sekumpulan data operasional perlu dilakukan sebelum tahap penggalian informasi dalam KDD dimulai. Data hasil seleksi yang akan digunakan untuk proses *data mining*, disimpan dalam suatu berkas, terpisah dari basis data operasional.

2. *Pre-processing/Cleaning*

Sebelum proses *data mining* dapat dilaksanakan, perlu dilakukan proses *cleaning* pada data yang menjadi fokus KDD. Proses *cleaning* mencakup antara lain membuang duplikasi data, memeriksa data yang inkonsisten, dan memperbaiki kesalahan pada data, seperti kesalahan cetak (*tipografi*). Juga dilakukan proses *enrichment*, yaitu proses “memperkaya” data yang sudah

dengan data atau informasi lain yang relevan dan diperlukan untuk KDD, seperti data atau informasi eksternal.

### 3. *Transformation*

*Coding* adalah proses transformasi pada data yang telah dipilih, sehingga data tersebut sesuai untuk proses *data mining*. Proses *coding* dalam KDD merupakan proses kreatif dan sangat tergantung pada jenis atau pola informasi yang akan dicari dalam basis data.

### 4. *Data mining*

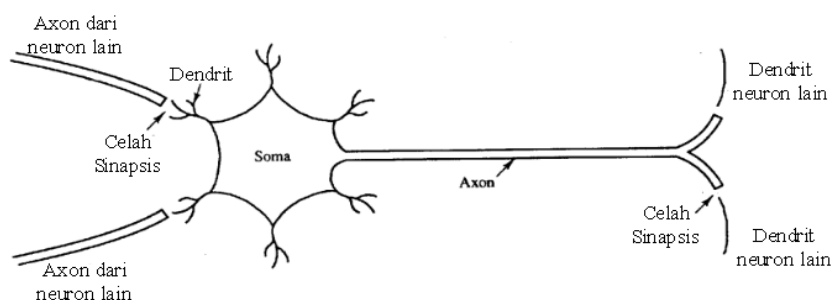
*Data mining* adalah proses mencari pola atau informasi menarik dalam data terpilih dengan menggunakan teknik atau metode tertentu. Teknik, metode, atau algoritma yang tepat sangat bergantung pada tujuan dan proses KDD secara keseluruhan.

### 5. *Interpretation/Evaluation*

Pola informasi yang dihasilkan dari proses *data mining* perlu ditampilkan dalam bentuk yang mudah dimengerti oleh pihak yang berkepentingan. Tahap ini mencakup pemeriksaan apakah pola atau informasi yang ditemukan bertentangan dengan fakta atau hipotesis yang ada sebelumnya.

## 2.2 Jaringan Syaraf Tiruan (JST)

Otak manusia memiliki struktur yang sangat kompleks dan memiliki kemampuan yang luar biasa. Otak terdiri dari neuron-neuron dan penghubung yang disebut sinapsis. Neuron bekerja berdasarkan implus atau sinyal yang diberikan pada neuron. Neuron meneruskannya pada neuron lain. Diperkirakan manusia memiliki  $10^2$  neuron dan  $6 \times 10^{18}$  sinapsis (Siang, 2005).



**Gambar 2.1** Model neuron biologis

Neuron memiliki 3 komponen penting yaitu dendrit, soma dan axon. Dendrit menerima sinyal dari neuron lain. Sinyal tersebut berupa impuls elektrik yang dikirim melalui celah sinaptik melalui proses kimiawi. Sinyal tersebut dimodifikasi (diperkuat/diperlemah) di celah sinaptik. Berikutnya, soma menjumlahkan semua sinyal-sinyal yang masuk. Kalau jumlahan tersebut cukup kuat dan melebihi batas ambang (threshold), maka sinyal tersebut akan diteruskan ke sel lain melalui axon. Frekuensi penerusan sinyal berbeda-beda antara satu sel dengan yang lain.

Jaringan syaraf tiruan adalah system pemrosesan informasi yang memiliki karakteristik mirip dengan jaringan syaraf biologi. Jaringan syaraf tiruan dikembangkan sebagai model matematika dari pengetahuan manusia atau syaraf biologi, berdasarkan asumsi bahwa (Siang, 2005):

1. Pemrosesan informasi terjadi pada elemen yang disebut neuron.
2. Sinyal dilewatkan antar neuron melalui link penghubung.
3. Setiap link penghubung memiliki yang dikalikan dengan sinyal yang dikirimkan.
4. Setiap neuron menggunakan fungsi aktivasi (biasanya non linier) pada jaringan *inputnya* (jumlah bobot sinyal *input*) untuk menghitung sinyal *output*.

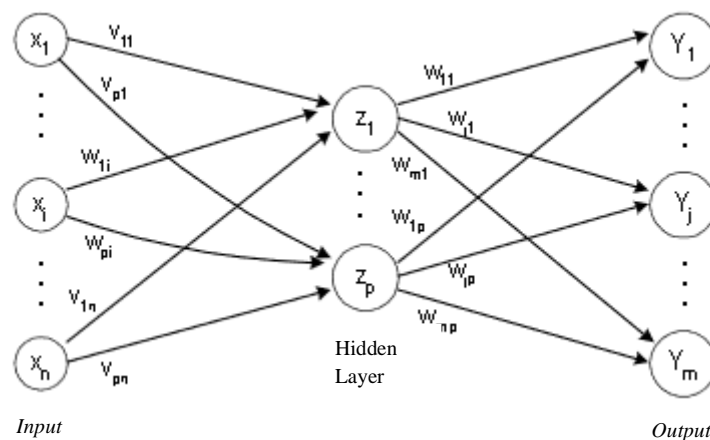
Jaringan Syaraf ditentukan oleh tiga hal, yaitu:

1. Pola jaringan antar neuron (arsitektur jaringan).
2. Metode untuk meramalkan bobot penghubung (metode *training/learning*/algoritma).
3. Fungsi aktivasi, untuk menentukan besarnya *output* dari suatu neuron.

Sebuah jaringan syaraf tiruan terdiri dari sejumlah elemen pemrosesan yaitu lapisan (*layer*) dan neuron (Gambar 2.2). Setiap neuron terhubung ke neuron lain melalui link penghubung dengan bobot tertentu. Jenis *layer* dapat dibedakan menjadi:

1. *Input layer*, terdiri dari unit-unit neuron yang berperan sebagai *input* proses pengolahan data pada jaringan syaraf.

2. *Hidden layer*, terdiri dari unit-unit neuron yang dianalogikan sebagai lapisan tersembunyi dan berperan sebagai lapisan yang meneruskan respon dari *input*. Pengaturan jumlah neuron pada *hidden layer* yang terlalu sedikit akan menyebabkan jaringan tidak cukup fleksibel untuk mempelajari, *error* atau kesalahan pembelajaran meningkat dan jumlah neuron yang terlalu banyak akan menyebabkan terjadinya *overfitting*.
3. *Output layer*, terdiri dari unit-unit neuron yang berperan memberikan solusi dari data *input*.



**Gambar 2.2** Arsitektur Jaringan Syaraf Tiruan Multi layer

Terdapat beberapa fungsi aktivasi yang sering digunakan dalam jaringan syaraf tiruan antara lain:

1. Fungsi *threshold* (Batas ambang)

$$f(x) = \begin{cases} 1 & \text{jika } x \geq a \\ 0 & \text{jika } x < a \end{cases} \quad (2.1)$$

Untuk beberapa kasus fungsi *threshold* yang dibuat tidak berharga 0 atau 1, tetapi -1 atau 1 (atau sering disebut *threshold bipolar*).

$$f(x) = \begin{cases} 1 & \text{jika } x \geq a \\ -1 & \text{jika } x < a \end{cases} \quad (2.2)$$

## 2. Fungsi sigmoid

$$f(x) = \frac{1}{1+e^{-x}} \quad (2.3)$$

Fungsi sigmoid sering dipakai karena nilai fungsinya yang terletak antara 0 dan 1 serta dapat diturunkan dengan mudah.

$$f'(x) = f(x)(1 - f(x)) \quad (2.4)$$

## 3. Fungsi Identitas

$$f(x) = x \quad (2.5)$$

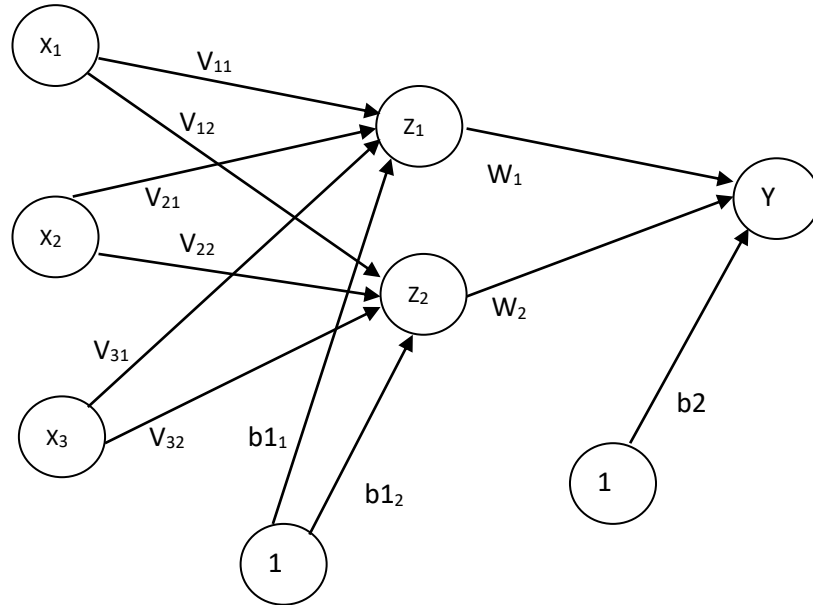
Fungsi identitas sering dipakai apabila kita menginginkan keluaran jaringan berupa sembarang bilangan riil, bukan hanya pada *range* [0,1] atau [-1,1].

### 2.2.1 Jaringan Saraf Tiruan *Backpropagation*

Algoritma pembelajaran jaringan syaraf tiruan *Backpropagation* pertama kali dirumuskan oleh Werbos dan di populerkan oleh Rumelhart dan McClelland. Jaringan syaraf *Backpropagation* merupakan algoritma pembelajaran terbimbing yang mempunyai banyak lapisan. *Backpropagation* menggunakan *error output* untuk menggunakan nilai bobot-bobotnya dalam arah mundur (*backward*). Untuk mendapatkan *error* ini, tahapan perambatan maju (*forward propagation*) harus dikerjakan terlebih dahulu.

Arsitektur jaringan *Backpropagation* pada Gambar 2.3 terlihat jaringan terdiri atas 3 unit (neuron) pada lapisan *input*, yaitu  $X_1$ ,  $X_2$ , dan  $X_3$ ; 1 lapisan tersembunyi dengan dua neuron, yaitu  $Z_1$  dan  $Z_2$  serta satu unit pada lapisan *output*, yaitu  $Y$ . Bobot yang menghubungkan  $X_1$ ,  $X_2$ , dan  $X_3$  dengan neuron pertama pada lapisan tersembunyi, adalah  $V_{11}$ ,  $V_{21}$ , dan  $V_{31}$  ( $V_{ij}$ : bobot yang menghubungkan neuron *input* ke- $i$  ke neuron ke- $j$  pada lapisan tersembunyi).  $b_{11}$  dan  $b_{12}$  adalah bobot bias yang menuju ke neuron pertama dan kedua pada lapisan tersembunyi. Bobot yang menghubungkan  $Z_1$  dan  $Z_2$  dengan neuron pada lapisan *output*, adalah  $W_1$  dan  $W_2$ . Bobot bias  $b_2$  menghubungkan lapisan tersembunyi

dengan lapisan *output*. Fungsi aktivasi yang digunakan, antara lapisan *input* dan lapisan tersembunyi.



**Gambar 2.3** Arsitektur *Backpropagation*

Berikut ini merupakan algoritma dalam pembelajaran *Backpropagation* (Kusumadewi, 2014):

0. Inisialisasi bobot (diberikan nilai random yang berdistribusi normal).
1. Ketika kondisi berhenti tidak terpenuhi, dilakukan langkah 2
2. Untuk tiap tiap pasangan elemen yang akan dilakukan pembelajaran, kerjakan langkah-langkah berikutnya.

***Feedforward:***

- a. Tiap-tiap unit *input* ( $X_i, i = 1, 2, \dots, n$ ) menerima sinyal  $X_i$  dan meneruskan sinyal tersebut ke semua unit lapisan tersembunyi.
- b. Tiap-tiap unit pada suatu lapisan tersembunyi ( $Z_j, j = 1, 2, \dots, p$ ) menjumlahkan sinyal sinyal *input* terbobot,

$$z\_in_j = b1_j + \sum_{i=1}^n x_i v_{ij}, \quad (2.6)$$

menerapkan fungsi aktivasi untuk menghitung sinyal *output*,

$$z_j = f(z\_in_j), \quad (2.7)$$

dan mengirimkan sinyal *output* tersebut sebagai *input* untuk semua unit lapisan *output*.

- c. Tiap-tiap *output*  $Y_k (k = 1, 2, \dots, m)$  menjumlahkan sinyal-sinyal *input* yang terboboti,

$$y\_in_k = b2_k + \sum_{i=1}^p z_i w_{jk}, \quad (2.8)$$

menerapkan fungsi aktivasi untuk menghitung sinyal *output*,

$$y_k = f(y\_in_k), \quad (2.9)$$

mengirimkan sinyal tersebut ke semua unit lapisan atasnya (unit-unit *output*).

Catatan : langkah 2 (b) dilakukan sebanyak jumlah lapisan tersembunyi.

***Backpropagation:***

- d. Tiap-tiap unit *output*  $Y_k (k = 1, 2, \dots, m)$  menerima target pola yang berhubungan dengan pola *input* pelatihan, kemudian dihitung informasi *error*nya,

$$\delta 2_k = (t_k - y_k) f'(y\_in_k), \quad (2.10)$$

menghitung besar koreksi bobot yang digunakan untuk mengupdate informasi *error*nya,

$$\Delta w_{jk} = \alpha \delta_k z_j, \quad (2.11)$$

kemudian menghitung koreksi bias yang digunakan untuk mengupdate bias  $b2_k$ ,

$$\Delta b2_k = \alpha \delta_k, \quad (2.12)$$

dan mengirim  $\delta_k$  ke unit lapisan tersembunyi.

Langkah 2 (d) dilakukan sebanyak jumlah lapisan tersembunyi, yaitu menghitung informasi error dari suatu lapisan tersembunyi ke lapisan tersembunyi sebelumnya.

- e. Tiap-tiap unit tersembunyi  $Z_j (j = 1, 2, \dots, p)$  menjumlahkan delta *input*nya dari unit-unit lapisan *output*,

$$\delta\_in_j = \sum_{k=1}^m \delta 2_k w_{jk}, \quad (2.13)$$

kalikan nilai ini dengan turunan dari fungsi aktivasinya untuk menghitung *error*,



$$\delta 1_j = \delta_{in_j} f'(z_{in_j}) \quad (2.14)$$

menghitung besar koreksi bobot yang digunakan untuk mengupdate bobot  $v_{ij}$ ,

$$\Delta v_{ij} = \alpha \delta 1_j x_j \quad (2.15)$$

menghitung koreksi bias yang digunakan untuk mengupdate bias  $b1_j$ ,

$$\Delta b1_j = \alpha \delta 1_j \quad (2.16)$$

- f. Tiap-tiap unit *output* ( $Y_k, k = 1, 2, 3, \dots, m$ ) mengupdate bias dan bobotnya ( $j = 1, 2, \dots, p$ ),

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta w_{jk} \quad (2.17)$$

$$b2_k(\text{baru}) = b2_k(\text{lama}) + \Delta b2_k \quad (2.18)$$

Tiap-tiap unit tersembunyi ( $Z_j, j = 1, 2, 3, \dots, p$ ) mengupdate bias dan bobotnya ( $i = 0, 1, 2, \dots, n$ ),

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij} \quad (2.19)$$

$$b1_j(\text{baru}) = b1_j(\text{lama}) + \Delta b1_j \quad (2.20)$$

3. Uji kondisi berhenti.

### 2.3 Algoritma Genetika

Algoritma Genetika merupakan suatu teknik optimasi yang didasarkan pada prinsip genetik dan seleksi alam. Dalam algoritma Genetika populasi terbentuk dari banyak individu yang berkembang sesuai aturan seleksi spesifik dengan memaksimalkan fitness (Haupt & Haupt, 2004). Algoritma ini juga digunakan untuk mendapatkan nilai global optimum dengan cara melakukan perulangan atau iterasi pada konsep evolusi Darwin.

Pemetaan proses alamiah ke dalam proses komputasi Algoritma Genetika dapat dirangkum dalam Tabel 2.1. (Zukhri, 2014)

**Tabel 2.1** Pemetaan proses alamiah ke dalam proses komputasi GA

<b>Proses Alamiah</b>	<b>Proses Komputasi</b>
Individu	Penyelesaian masalah
Populasi	Himpunan Penyelesaian
Fitness/kebugaran	Kualitas penyelesaian
Kromosom	Kode/representasi penyelesaian
Gen	Bagian dari representasi penyelesaian
Pertumbuhan	Pendekodean representasi penyelesaian
Penyilangan	Operator genetika
Mutasi	Operator genetika
Seleksi Alam	Menyelesaikan penyelesaian masalah (sementara) berdasarkan kualitasnya

Menurut Trevino & Falciani (2006) terdapat 7 tahapan untuk menjalankan algoritma genetika, yaitu :

1. Membentuk Populasi awal terdiri dari beberapa kromosom yang didalamnya memuat gen kromosom pada algoritma genetika yang digunakan untuk menunjukkan kandidat sekelompok gen yang dapat digunakan sebagai solusi permasalahan.
2. Masing-masing kromosom dalam populasi dievaluasi kemampuannya dengan menggunakan fungsi fitness. Pada penelitian ini fungsi *fitness* yang digunakan adalah nilai ketepatan klasifikasi. Sehingga individu yang akan bertahan kegenerasi selanjutnya adalah individu yang memiliki nilai *fitness* tertinggi.
3. Ketika sebuah kromosom memiliki nilai fitness lebih optimum daripada nilai inisialnya, maka kromosom dihentikan, namun apabila tidak maka tahapan analisis dilanjutkan ke tahap 4. Nilai fitness tertinggi yang terpilih sebagai solusi permasalahan dari penelitian ini, karena fungsi fitness yang digunakan adalah tingkat ketepatan klasifikasi.
4. Memilih kromosom dengan nilai fitness yang optimum yang dijadikan orang tua.

5. Mengkombinasikan informasi genetika yang ada dalam replikasi orang tua melalui pindah silang ( $P_c$ ). Dua induk secara random dipilih dan digunakan untuk membentuk dua kromosom baru dilakukan elitism, yaitu prosedur pengopian individu agar individu yang bernilai fitness tertinggi tidak hilang selama proses evolusi.
6. Melakukan mutasi ( $P_m$ ) untuk memperkenalkan unsur gen baru pada kromosom secara acak.
7. Tahapan diulangi dari tahapan 2 sampai kromosom yang memberikan nilai fitness paling optimum atau sudah mencapai konvergen.

### 2.3.1 Pengkodean

Pengkodean merupakan bagian penting dalam Algoritma genetika. Proses ini diperlukan dalam kaitannya dengan peranan kromosom sebagai representasi penyelesaian masalah (Zukhri, 2014). Pengkodean adalah proses menggambarkan bentuk gen dalam kromosom. Pengkodean dapat berupa *binary encoding*, *real number encoding*, *discreate encoding*, *value encoding*, dan lain lain.

### 2.3.2 Fitness

Fungsi Fitness digunakan untuk mengukur tingkat kebaikan atau kesesuaian suatu solusi yang dicari. Fungsi fitness bisa berhubungan langsung dengan fungsi tujuan. Sejumlah solusi yang dibangkitkan dalam populasi akan dievaluasi menggunakan fungsi fitness. Fungsi fitness ( $F(u)$ ) yang digunakan adalah:

$$F(u) = f(u) \quad (2.21)$$

Dimana  $f(u)$  adalah fungsi tujuan dari masalah yang dapat terselesaikan (Santosa & Willy, 2011).

### 2.3.3 Seleksi Orang Tua (*Parents*)

Seleksi orang tua bertujuan untuk memberikan kesempatan reproduksi bagi anggota populasi yang memiliki nilai fitness tinggi (Desiani & Arhami, 2006). Pemilihan dua buah kromosom dalam suatu populasi sebagai orang tua yang akan dipindah silangkan biasanya secara proporsional sesuai dengan nilai

fitness masing masing. Metode umum yang dipakai adalah Roulette Wheel (Roda Roulette). Pada metode ini, masing masing kromosom menempati potongan lingkaran pada roda Roulette secara proporsional sesuai nilai fitnessnya. Sebuah kromosom yang nantinya akan terpilih adalah apabila bilangan random yang dibangkitkan berada dalam nilai interval kumulatifnya. Nilai kumulatif ini didapatkan dari membagi nilai fitness dari tiap kromosom dengan total nilai fitness keseluruhan (Suyanto, 2005).

#### **2.3.4 Pindah Silang**

Pindah silang merupakan metode pengkombinasian 2 orang tua kromosom untuk membentuk generasi baru. Orang tua diperoleh dari tahapan seleksi orang tua menggunakan *Roulette wheel selection* (RWS). Kromosom dengan nilai fitness yang tinggi akan dijadikan sebagai orang tua.

#### **2.3.5 Mutasi**

Metode yang digunakan untuk mutasi pada penelitian ini adalah metode uniform. Tujuan mutasi adalah untuk mendapatkan keberagaman gen. Terdapat 2 tahapan dalam metode ini, Pertama adalah memilih sebagian gen dari kromosom yang akan dimutasi dengan peluang sebesar 0.1, kedua mengganti gen yang terpilih dengan bilangan random yang dibangkitkan dari batas bawah dan batas atas nilai gen dalam kromosom.

#### **2.3.6 Elitisme**

Suatu individu yang memiliki nilai fitness tertinggi tidak akan selalu terpilih karena proses seleksi dilakukan secara random. Oleh karena itu perlu dilakukan elitism yaitu suatu prosedur pengopian individu agar individu yang bernilai fitness tinggi tidak hilang selama proses evolusi (Suyanto, 2005).

#### **2.3.7 Klasifikasi**

Klasifikasi merupakan suatu pekerjaan menilai objek data untuk memasukkannya kedalam kelas tertentu dari sejumlah kelas yang tersedia. Dalam klasifikasi ada dua pekerjaan utama yang dilakukan, yaitu: Pertama,

pembangunan model sebagai prototype untuk disimpan sebagai memori dan kedua, penggunaan model tersebut untuk melakukan pengenalan/ klasifikasi/ prediksi pada suatu objek data tersebut dalam model yang mudah disimpan.

Sebuah system yang melakukan klasifikasi diharapkan dapat melakukan klasifikasi semua set data dengan benar, tetapi tidak dapat dipungkiri bahwa kinerja suatu sistem tidak bisa 100% benar sehingga sebuah system klasifikasi juga harus diukur kinerjanya. Umumnya, pengukuran kinerja dilakukan dengan matriks konfusi.

## 2.4 Evaluasi Performansi Metode Klasifikasi

Data aktual dan data hasil prediksi dari model klasifikasi disajikan dengan menggunakan tabulasi silang (confusion matrix), yang mengandung informasi tentang kelas data aktual direpresentasikan pada baris matriks dan kelas data hasil prediksi pada kolom (Han & Kamber, 2006). Dimana untuk Confusion Matrix dapat dilihat pada Tabel 2.2

**Tabel 2.2** Confusion Matriks

Aktual	Prediksi	
	Negatif	Positif
Negatif	TN	FP
Positif	FN	TP

Keterangan :

1. True Positve (TP) menunjukan bahwa kelas yang dihasilkan prediksi klasifikasi adalah positif dan kelas sebenarnya adalah positif.
2. True Negativ (TN) menunjukan bahwa kelas yang dihasilkan prediksi klasifikasi adalah negative dan kelas sebenarnya negativ.
3. False Positif (FP) menunjukan bahwa kelas yang dihasilkan prediksi klasifikasi adalah negative dan kelas sebenarnya positif.
4. False Negativ (FN) menunjukan bahwa kelas yang dihasilkan prediksi klasifikasi adalah positif dan kelas sebenarnya negativ.

Ketepatan klasifikasi dapat dilihat dari akurasi klasifikasi. Akurasi klasifikasi menunjukkan performansi model klasifikasi secara keseluruhan, dimana semakin tinggi akurasi klasifikasi hal ini berarti semakin baik performansi model klasifikasi.

$$\text{Akurasi Total} = \frac{TP+TN}{TP+FP+TN+FN} \times 100\% \quad (2.22)$$

Untuk mendapatkan klasifikasi yang optimal dan lebih spesifikasi maka dapat diuji *Sensitivity* dan *Specificity*. *Sensitivity* adalah prosentase data positif yang diprediksi sebagai positif sedangkan *Specificity* adalah prosentase data negative diprediksi sebagai negatif.

$$\text{Sensitivity} = \frac{TP}{(TP+FN)} \times 100\% \quad (2.23)$$

$$\text{Specificity} = \frac{TN}{(TN+FP)} \times 100\% \quad (2.24)$$

Evaluasi performansi model klasifikasi dapat dilakukan dengan menggunakan *G-mean*. *G-mean* merupakan rata-rata *geometric Sensitivity* dan *Specificity*. *G-mean* akan bernilai nol jika kelas positif tidak dapat diprediksi.

$$G - \text{Mean} = \sqrt{\text{Sensitivity} \times \text{Specificity}} \quad (2.25)$$

Evaluasi performansi model klasifikasi dapat juga dilakukan dengan menggunakan AUC (*Area Under Curve*).

$$\text{AUC} = \frac{1+TPR-FPR}{2} \quad (2.26)$$

dimana *true positive rate* (TPR) dan *false positive rate* (FPR),

$$\text{TPR} = \text{Sensitivity} = \frac{TP}{(TP+FN)} \quad (2.27)$$

$$\text{FPR} = 1 - \text{Specificity} = 1 - \frac{TN}{(TN+FP)} \quad (2.28)$$

## 2.5 Cross Validation

*Cross-validation* adalah sebuah pendekatan yang digunakan untuk mengestimasi ketepatan klasifikasi dari sebuah model dan didasarkan pada pemisahan sampel yang tersedia antara data pembelajaran (*training*) dan data validasi (*testing*) (Last, 2006). Bentuk dasar dari *cross-validation* adalah *k-foldcross-validation*. Dalam *k-foldcross-validation*, pertama data dipartisi menjadi *k* bagian atau *fold* yang sama. Selanjutnya, *k* iterasi pembelajaran dan validasi

dilakukan sedemikian rupa sehingga dalam setiap iterasi sebuah *fold* berbeda digunakan untuk validasi sedangkan *k-1fold* sisanya digunakan untuk pembelajaran (*training*).

Selanjutnya, masalah yang penting untuk menjadi perhatian adalah memilih nilai  $k$  yang tepat. Biasanya nilai  $k$  besar lebih diinginkan karena memberikan perkiraan kinerja yang lebih baik dan ukuran data *training* lebih mendekati ukuran populasi, sehingga meningkatkan kemungkinan bahwa kesimpulan yang dibuat tentang algoritma pembelajaran yang diuji akan memberikan generalisasi untuk kasus dimana semua data digunakan untuk pembelajaran. Meningkatkan  $k$ , mengakibatkan tumpang tindih, data pembelajaran meningkat. Selain itu, meningkatkan  $k$  mengurangi ukuran data uji dan mengarah pada berkurangnya akurasi. Dalam Data mining  $k = 10$  dianggap sangat tepat karena menghasilkan prediksi menggunakan 90% dari data, sehingga akan memberikan generalisasi data secara lengkap.

Kohavi (1995), membandingkan beberapa pendekatan untuk menghitung akurasi. Diantaranya *cross-validation* (*regular cross-validation*, *leave-one-out cross-validation* dan *stratified cross-validation*) dan *bootstrap* (sampel dengan pengembalian), dan merekomendasikan *10-fold cross-validation* sebagai metode pemilihan model terbaik, karena cenderung memberikan akurasi sebagai bias yang lebih sedikit.

## 2.6 Bidikmisi

Bidikmisi adalah bantuan biaya pendidikan dari Kementerian Riset Teknologi dan Pendidikan Tinggi Republik Indonesia yang memberikan fasilitas pembebasan biaya pendidikan dan subsidi biaya hidup. Bidikmisi diberikan kepada penerima selama 8 (delapan) semester untuk S1 / D4, 6 (enam) semester untuk D3, 4 (empat) semester untuk D2, dan 2 (dua) semester untuk D1. (Panduan Bidikmisi, 2017).

### **2.6.1 Misi**

1. Menghidupkan harapan bagi masyarakat tidak mampu secara ekonomi namun mempunyai potensi akademik baik untuk dapat menempuh pendidikan sampai ke jenjang pendidikan tinggi.
2. Memberikan akses bagi masyarakat kurang mampu tapi memiliki potensi akademik yang baik untuk menjadi sumber daya manusia yang memiliki nilai-nilai kebangsaan, patriotism, cinta Tanah Air, dan semangat bela negara.
3. Memberikan kesempatan bagi masyarakat kurang mampu tapi memiliki potensi akademik yang baik untuk ikut berperan serta dalam meningkatkan daya saing bangsa di era kompetisi global, khususnya dalam menghadapi Masyarakat Ekonomi ASEAN (MEA) yang telah diratifikasi oleh seluruh Negara ASEAN.

### **2.6.2 Tujuan**

1. Meningkatkan akses dan kesempatan belajar di perguruan tinggi bagi peserta didik yang tidak mampu secara ekonomi dan berpotensi akademik baik.
2. Memberi bantuan biaya pendidikan kepada calon /mahasiswa yang memenuhi kriteria untuk menempuh pendidikan program Diploma/Sarjana sampai selesai dan tepat waktu.
3. Meningkatkan prestasi mahasiswa, baik pada bidang kurikuler, ko-kurikuler maupun ekstra kurikuler.
4. Menimbulkan dampak iring bagi mahasiswa dan calon mahasiswa lain untuk selalu meningkatkan prestasi dan kompetif.
5. Melahirkan lulusan yang mandiri, produktif dan memiliki kepedulian sosial, sehingga mampu berperan dalam upaya pemutusan mata rantai kemiskinan dan pemberdayaan masyarakat.

### **2.6.3 Sasaran**

Sasaran program adalah lulusan satuan pendidikan SMA/SMK/MA atau bentuk lain yang sederajat tahun 2016 dan 2017 yang tidak mampu secara ekonomi dan memiliki potensi akademik baik.



#### **2.6.4 Persyaratan Dan Kuota**

##### **A. Persyaratan Calon Penerima**

Persyaratan untuk mendaftar tahun 2017 adalah sebagai berikut:

1. Siswa SMA/SMK/MA/MAK atau bentuk lain yang sederajat yang akan lulus pada tahun 2017.
2. Lulusan tahun 2016 yang bukan penerima Bidikmisi dan tidak bertentangan dengan ketentuan penerimaan mahasiswa baru di masing-masing perguruan tinggi.
3. Usia paling tinggi pada saat mendaftar adalah 21 tahun.
4. Tidak mampu secara ekonomi dengan kriteria:
  - a. Siswa penerima Beasiswa Siswa Miskin (BSM).
  - b. Pemegang Kartu Indonesia Pintar (KIP) atau sejenisnya.
  - c. Pendapatan kotor gabungan orangtua/wali (suami istri) sebesar-besarnya Rp3.000.000,00 per bulan. Untuk pekerjaan non formal/informal pendapatan yang dimaksud adalah rata-rata penghasilan per bulan dalam satu tahun terakhir; dan atau
  - d. Pendapatan kotor gabungan orangtua/wali dibagi jumlah anggota keluarga sebesar-besarnya Rp750.000,00 setiap bulannya.
5. Pendidikan orang tua/wali setinggi-tingginya S1 (Strata 1) atau Diploma 4.
6. Berpotensi akademik baik berdasarkan rekomendasi kepala sekolah.
7. Pendaftar difasilitasi untuk memilih salah satu diantara PTN atau PTS dengan ketentuan:
  - a. PTN dengan pilihan seleksi masuk:
    - 1) Seleksi Masuk Perguruan Tinggi Negeri (SNMPTN).
    - 2) Seleksi Bersama Masuk Perguruan Tinggi Negeri (SBMPTN).
    - 3) Seleksi mandiri PTN.
  - b. Politeknik, UT, dan Institut Seni dan Budaya.
  - c. PTS sesuai dengan pilihan masuk.

## **B. Kuota Mahasiswa Baru**

1. Kuota Bidikmisi diperuntukkan bagi mahasiswa yang lulus:
  - a. SNMPTN.
  - b. SBMPTN.
  - c. Seleksi mandiri PTN.
  - d. Seleksi di Politeknik, UT, dan Institut Seni dan Budaya.
  - e. Seleksi di PTS
2. Kuota awal Bidikmisi bagi PTN, Politeknik dan Institut Seni ditetapkan dengan mempertimbangkan kondisi PTN kondisi umum ekonomi mahasiswa, kapasitas daya tampung PTN, dan alokasi kuota setahun sebelumnya.
3. Kuota PTS melalui seleksi mandiri ditetapkan oleh Kopertis berdasarkan: (1) jumlah program studi yang memenuhi persyaratan akreditasi, dengan proporsi maksimal 20% dari total mahasiswa baru; (2) Kondisi geografis, karakteristik sosial ekonomi sekitar perguruan tinggi negeri untuk kekhususan wilayah 3T; (3) ketaatan perguruan tinggi terhadap azas pengelolaan yang baik.
4. Kuota untuk PTS termasuk penentuan program studi dilakukan oleh Ditjen Belmawa bersama Kopertis Wilayah dengan kriteria khusus.

### **2.6.5 Mekanisme Seleksi**

#### **2.6.5.1 Sosialisasi dan Koordinasi**

1. Kementerian Riset Teknologi dan Pendidikan Tinggi melakukan koordinasi dan sosialisasi antar unit utama, unit kerja dan instansi terkait termasuk Panitia Seleksi Nasional mahasiswa baru serta melakukan publikasi melalui media massa.
2. Dinas pendidikan provinsi dan kabupaten/kota melakukan sosialisasi dan atau memberikan informasi kepada satuan pendidikan di lingkungannya tentang program Bidikmisi.
3. Institusi pendidikan tinggi melakukan sosialisasi dan atau memberikan informasi kepada sekolah dan publik tentang program Bidikmisi.
4. Kepala Sekolah/Madrasah/PKBM atau yang sederajat mensosialisasikan program Bidikmisi kepada siswa khususnya bagi siswa kelas 12.

5. Kepala Sekolah/Madrasah/PKBM atau yang sederajat mengoordinasikan dan memfasilitasi seluruh proses pendaftaran di setiap sekolah dan mengirimkan berkas yang telah memenuhi persyaratan ke perguruan tinggi negeri yang dituju tanpa mengenakan biaya pada siswa pendaftar.

#### **2.6.5.2 Pendaftaran**

##### **A. Pendaftaran Daring (*On-line*)**

Tata cara pendaftaran Bidikmisi melalui SNMPTN, SBMPTN, PMDK Politeknik atau Seleksi Mandiri perguruan tinggi secara daring pada laman bidikmisi (<http://bidikmisibelmawa.ristekdikti.go.id/>) adalah sebagai berikut:

1. Tahapan pendaftaran Bidikmisi
  - a. Sekolah mendaftarkan diri sebagai instansi pemberi rekomendasi ke laman Bidikmisi dengan melampirkan hasil pindaian (scan) untuk mendapatkan nomor Kode Akses Sekolah.
  - b. Ditjen Belmawa memverifikasi pendaftaran dalam kurun waktu 1 x 24 jam pada hari dan jam kerja.
  - c. Sekolah merekomendasikan masing-masing siswa melalui laman Bidikmisi menggunakan kombinasi NPSN dan Kode akses yang telah diverifikasi.
  - d. Sekolah memberikan nomor pendaftaran dan kode akses kepada masing-masing siswa yang sudah direkomendasikan.
  - e. Siswa mendaftar melalui laman Bidikmisi dan menyelesaikan semua tahapan yang diminta didalam sistem pendaftaran.
2. Siswa yang sudah menyelesaikan pendaftaran bidikmisi mendaftar seleksi nasional atau mandiri yang telah diperoleh sesuai ketentuan masing-masing pola seleksi melalui alamat berikut.
  - a. SNMPTN melalui [http:// snmptn.ac.id](http://snmptn.ac.id)
  - b. SBMPTN melalui <http://sbmptn.ac.id>.
  - c. PMDK Politeknik melalui <http://pmdk.politeknik.or.id>.

- d. Seleksi Mandiri PTN sesuai ketentuan masing-masing PTN.
- e. Seleksi Mandiri PTS sesuai ketentuan masing masing PTS.

Siswa yang mendaftar dan ditentukan lolos melalui seleksi masuk, melengkapi berkas dan dibawa pada saat pendaftar ulang.

#### **B. Pendaftaran Langsung (*Off-line*)**

1. Sekolah dan atau calon yang tidak dapat melakukan tahapan pendaftaran Bidikmisi secara *on-line* untuk Seleksi Mandiri karena keterbatasan akses internet, maka:
  - a. Calon mengisi formulir yang disediakan oleh sekolah (formulir dapat diunduh di [www.dikti.go.id](http://www.dikti.go.id) atau [www.bidikmisi.dikti.go.id](http://www.bidikmisi.dikti.go.id)) dan selanjutnya formulir yang telah diisi beserta berkas persyaratan lainnya disampaikan ke Kepala Sekolah.
  - b. Kepala Sekolah mengirimkan formulir rekomendasi, formulir pendaftaran berserta kelengkapan berkas lainnya secara kolektif kepada masing-masing Rektor/Direktur/Ketua PTN yang menyelenggarakan seleksi mandiri masuk perguruan tinggi negeri sesuai pilihan calon.

#### **2.6.5.3 Jenis Seleksi dan Metode Verifikasi**

Perguruan Tinggi dapat melakukan seleksi Bidikmisi melalui seleksi nasional maupun seleksi mandiri.

#### **I. Seleksi Untuk Perguruan Tinggi Negeri**

##### **A. Seleksi Nasional / Bersama**

- 1) PTN melakukan seleksi terhadap penerima rekomendasi Bidikmisi yang merupakan lulusan seleksi nasional (SNMPTN-SBMPTN) sesuai persyaratan dan kriteria yang ditetapkan oleh masing-masing PTN.
- 2) Seleksi ditentukan oleh masing-masing PTN dengan memprioritaskan, pendaftar yang mempunyai potensi akademik yang paling tinggi, pendaftar yang paling tidak mampu secara ekonomi, urutan kualitas Sekolah, dan memperhatikan asal daerah pendaftar. Untuk memastikan

kondisi ekonomi pendaftar, akan lebih baik kalau PTN melakukan kunjungan ke alamat pendaftar.

- 3) Pertimbangan khusus dalam kelulusan seleksi diberikan kepada pendaftar yang mempunyai prestasi ekstrakurikuler paling rendah peringkat ke-3 di tingkat kabupaten/kota atau prestasi non kompetitif lain yang tidak ada pemeringkatan (contoh ketua organisasi siswa sekolah/OSIS).
- 4) Pelamar Bidikmisi penerima BSM dan/atau memiliki KIP dan sejenisnya dapat dikecualikan dalam proses verifikasi kelayakan ekonomi. Namun, jika di kemudian hari ditemukan ternyata tidak layak dapat dikenakan sanksi.
- 5) Kunjungan ke alamat pendaftar dapat dilakukan dengan mendayagunakan mahasiswa PTN yang bersangkutan atau PTN dari domisili pendaftar dengan mekanisme yang disetujui bersama.
- 6) Hasil seleksi nasional calon mahasiswa diumumkan oleh pengelola di tingkat perguruan tinggi dan diinformasikan ke Ditjen Belmawa melalui Sistem Informasi Manajemen Bidikmisi.

## **B. Seleksi Mandiri (Seleksi Lokal)**

PTN dapat melakukan seleksi Bidikmisi melalui seleksi mandiri perguruan tinggi dengan ketentuan:

1. PTN melakukan seleksi terhadap pendaftar menggunakan jalur, persyaratan dan kriteria khusus yang ditetapkan oleh masing-masing PTN.
2. Seleksi ditentukan oleh masing-masing PTN dengan memprioritaskan pendaftar yang mempunyai potensi akademik yang paling tinggi, pendaftar yang paling tidak mampu secara ekonomi, urutan kualitas Sekolah, dan memperhatikan asal daerah pendaftar. Untuk memastikan kondisi ekonomi pendaftar, dianjurkan kalau PTN melakukan kunjungan ke alamat pendaftar. Disamping itu dapat juga dilakukan verifikasi dan rekomendasi oleh penerima Bidikmisi sebelumnya.
3. Pertimbangan khusus dalam kelulusan seleksi diberikan kepada pendaftar yang mempunyai prestasi ekstrakurikuler paling rendah peringkat ke-3

ditingkat kabupaten/kota atau prestasi non kompetitif lain yang tidak ada pemeringkatan (contoh ketua organisasi siswa sekolah/OSIS).

4. Apabila diperlukan tes di lokasi yang memerlukan kehadiran fisik pendaftar, maka seluruh biaya untuk mengikuti proses seleksi mandiri termasuk biaya transportasi dan akomodasi ditanggung oleh PTN yang bersangkutan.
5. Pelamar Bidikmisi penerima BSM dan/atau memiliki KIP dan sejenisnya dapat dikecualikan dalam proses verifikasi kelayakan ekonomi. Namun jika di kemudian hari ditemukan ternyata tidak layak dapat dikenai sanksi.
6. Hasil seleksi calon mahasiswa diumumkan oleh Rektor/Direktur /Ketua atau yang diberi wewenang melalui media yang dapat diakses oleh setiap pendaftar dan diinformasikan ke Ditjen Belmawa melalui Sistem Informasi Manajemen Bidikmisi.

## **II. Seleksi Untuk PTS**

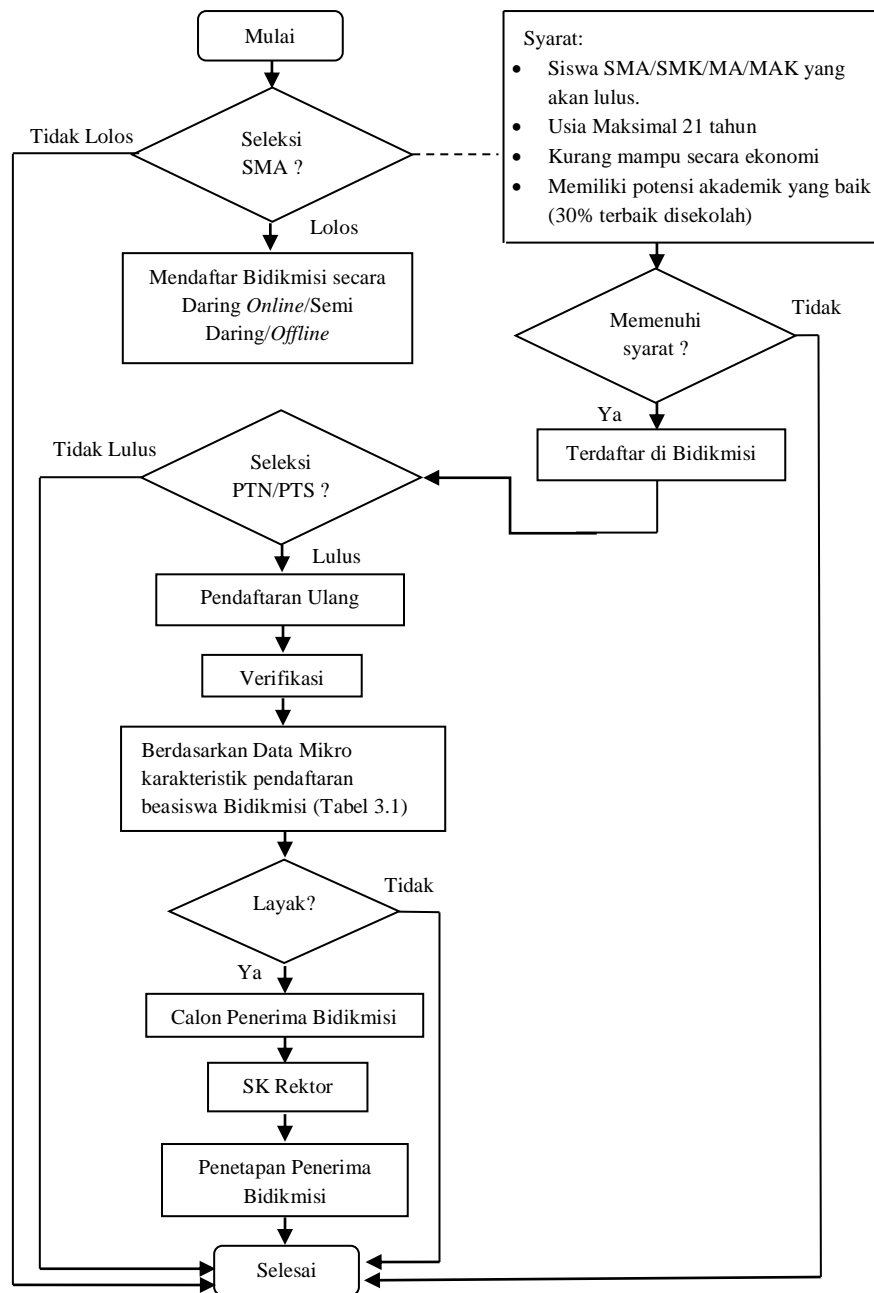
1. Seleksi ditentukan oleh masing-masing PTS dengan memprioritaskan pendaftar yang mempunyai potensi akademik yang paling tinggi, pendaftar yang paling tidak mampu secara ekonomi, dan memperhatikan asal daerah pendaftar. Untuk memastikan kondisi ekonomi pendaftar, akan lebih baik kalau PTS melakukan kunjungan ke alamat pendaftar.
2. Kunjungan ke alamat pendaftar dapat dilakukan dengan mendayagunakan mahasiswa PTS yang bersangkutan atau PTS dari domisili pendaftar dengan mekanisme yang disetujui bersama.
3. Pelamar Bidikmisi penerima BSM dan/atau memiliki KIP dan sejenisnya dapat dikecualikan dalam proses verifikasi kelayakan ekonomi. Namun, jika di kemudian hari ditemukan ternyata tidak layak dapat dikenai sanksi.
4. Hasil seleksi calon mahasiswa diumumkan oleh panitia seleksi PTS dan diinformasikan ke Ditjen Belmawa melalui Sistem Informasi Manajemen Bidikmisi.

#### **2.6.5.4 Pencalonan dan Penetapan**

Penetapan penerima bantuan biaya pendidikan Bidikmisi dilakukan melalui tata cara sebagai berikut:

1. Perguruan tinggi dapat melakukan koordinasi dengan PTN/PTS lain dari asal daerah pendaftar untuk melakukan visitasi/verifikasi.
2. Pelamar Bidikmisi penerima BSM dan/atau memiliki KIP dan sejenisnya dapat dikecualikan dalam proses verifikasi kelayakan ekonomi. Namun, jika di kemudian hari ditemukan ternyata tidak layak dapat dikenai sanksi; Sesuai pengumuman hasil seleksi nasional (SNMPTN-SBMPTN) dan Mandiri, maka calon mahasiswa melakukan daftar ulang di perguruan tinggi masing-masing.
3. Perguruan Tinggi melakukan pencalonan melalui SIM BIDIKMISI untuk pelamar Bidikmisi yang telah mendaftar ulang.
4. Pimpinan perguruan tinggi menerbitkan Surat Keputusan tentang Penetapan Penerima Bidikmisi untuk mahasiswa yang telah melakukan daftar ulang.
5. Perguruan tinggi melakukan penetapan calon menggunakan fasilitas SIM Bidikmisi.
6. Surat Keputusan dimaksud dikirimkan ke Ditjen Belmawa dan dilaporkan ke SIM Bidikmisi.

Diagram Alir secara umum tahapan proses seleksi penerima beasiswa Bidikmisi disajikan pada Gambar 2.4.



**Gamabar 2.4** Proses Seleksi Penerima Beasiswa Bidikmisi



*(halaman ini sengaja dikosongkan)*

## BAB 3

### METODOLOGI PENELITIAN

#### 3.1 Sumber Data dan Variabel Penelitian

Variabel penelitian yang digunakan dalam penelitian ini adalah data pendaftaran siswa Bidikmisi Tahun 2017. Data bersumber dari Database Kemenristek DIKTI kanal Bidikmisi. Variabel-variabel penelitian yang digunakan dalam penelitian ini berdasarkan formulir pendaftaran Program Bidikmisi. Variabel respon (Y) merupakan status beasiswa Bidikmisi seluruh kabupaten/kota di Provinsi Jawa Timur. Variabel prediktor (X) data mikro. Data mikro yaitu karakteristik calon mahasiswa pendaftaran beasiswa Bidikmisi di seluruh Kabupaten/Kota di Provinsi Jawa Timur. Rincian variabel respon dan variabel prediktor data mikro disajikan pada Tabel 3.1.

**Tabel 3.1** Deskripsi Variabel Penelitian

Variabel	Nama Variabel	Skala Data	Keterangan
Y	Status Penerimaan Beasiswa Bidikmisi	Nominal	0= Tidak Diterima Bidikmisi 1=Diterima Bidikmisi
X1	Pekerjaan Ayah	Nominal	1=Tidak Bekerja 2=Petani, Nelayan, Lainnya 3=TNI/POLRI 4=Wirausaha 5=Peg.Swasta 6=PNS
X2	Pekerjaan Ibu	Nominal	1=Tidak Bekerja 2=Petani, Nelayan, Lainnya 3=TNI/POLRI 4=Wirausaha 5=Peg.Swasta 6=PNS
X3	Pendidikan Ayah	Ordinal	1=Tidak Sekolah 2=Pendidikan Dasar (SD/MI dan SMP/MTs) 3=SMA/MA 4=PT (D1,D2,D3,D4/S1) 5=PT (S2,S3)

**Tabel 3.1** Deskripsi Variabel Penelitian (lanjutan)

Variabel	Nama Variabel	Skala Data	Keterangan
X4	Pendidikan Ibu	Ordinal	1=Tidak Sekolah 2=Pendidikan Dasar (SD/MI dan SMP/MTs) 3=SMA/MA 4=PT (D1,D2,D3,D4/S1) 5=PT (S2,S3)
X5	Penghasilan Ayah	Ordinal	1=Tidak Berpenghasilan 2=< Rp.1000.000 3=Rp.1000.000 - Rp.2000.000 4=Rp.2000.001- Rp.300.000 5= > Rp. 3000.001
X6	Penghasilan Ibu	Ordinal	1=Tidak Berpenghasilan 2=< Rp.1000.000 3=Rp.1000.000 - Rp.2000.000 4=Rp.2000.001- Rp.300.000 5= > Rp. 3000.001
X7	Kepemilikan Rumah Tinggal Keluarga	Nominal	1=Tidak Memiliki Rumah 2=Sewa(Tahunan, Bulanan), menumpang, dan menumpang tanpa ijin 3=Sendiri
X8	Sumber Listrik yang digunakan Keluarga	Nominal	1=Tidak Ada 2=Genset/Mandiri, Tenaga Surya 3=PLN
X9	Luas Tanah Rumah Tinggal Keluarga	Ordinal	1=25-50 m 2=50-9m 3=>100 m
X10	Luas Bangunan Rumah Tinggal Keluarga	Ordinal	1=25-50 m 2=50-99m 3=>100 m
X11	Kepemilikan Fasilitas Mandi Cuci Kakus	Nominal	1=Berbagi Pakai 2=Kepemilikan sendiri didalam 3=Kepemilikan sendiri diluar
X12	Jumlah Tanggungan	Nominal	1 =< 1 2=2 3=3 4=>4

Definisi Variabel-variabel yang digunakan dalam penelitian ini adalah sebagai berikut.

1. Status Penerimaan Beasiswa Bidikmisi

Status penerimaan siswa SLTA/SMA/MA kelas 12 yang mendaftar beasiswa Bidikmisi di Kabupaten/Kota Provinsi Jawa Tengah.

2. Pekerjaan Ayah

Pekerjaan Ayah adalah suatu aktivitas sehari-hari yang dilakukan Ayah untuk memenuhi kebutuhan harian.

3. Pekerjaan Ibu

Pekerjaan Ibu adalah suatu aktivitas sehari-hari yang dilakukan Ibu untuk memenuhi kebutuhan harian.

4. Pendidikan Ayah

Pendidikan Ayah adalah pendidikan formal tertinggi yang dimiliki oleh Ayah siswa pendaftaran beasiswa Bidikmisi.

5. Pendidikan Ibu

Pendidikan Ibu adalah pendidikan formal tertinggi yang dimiliki oleh Ibu siswa pendaftaran beasiswa Bidikmisi.

6. Penghasilan Ayah

Penghasilan Ayah adalah rata-rata penghasilan kotor perbulan yang diterima oleh Ayah/Wali dalam 12 bulan terakhir.

7. Penghasilan Ibu

Penghasilan Ibu adalah rata-rata penghasilan kotor perbulan yang diterima oleh Ibu dalam 12 bulan terakhir.

8. Kepemilikan Rumah Tinggal Keluarga

Kepemilikan Rumah Tinggal Keluarga adalah status kepemilikan rumah tinggal keluarga siswa yang mendaftar beasiswa Bidikmisi.

9. Sumber Listrik yang digunakan Keluarga

Sumber Listrik yang digunakan Keluarga adalah sumber listrik sehari-hari yang digunakan oleh rumah tinggal keluarga siswa pendaftar beasiswa Bidikmisi.

10. Luas Tanah Rumah Tinggal Keluarga

Luas Tanah Rumah Tinggal Keluarga adalah luas tanah yang ditempati dan digunakan oleh rumah tinggal keluarga siswa pendaftar beasiswa Bidikmisi. Luas tanah yang ditempati dan digunakan untuk keperluan sehari-hari (sebatas atap). Luas tanah dinyatakan dengan satuan meter persegi ( $m^2$ ).

11. Luas Bangunan Rumah Tinggal Keluarga

Luas Bangunan Rumah Tinggal Keluarga adalah luas bangunan yang ditempati dan digunakan oleh rumah tinggal keluarga siswa pendaftar beasiswa Bidikmisi. Luas tanah dinyatakan dengan satuan meter persegi ( $m^2$ ).

12. Kepemilikan Fasilitas Mandi Cuci Kakus

Kepemilikan Fasilitas Mandi Cuci Kakus adalah ketersediaan fasilitas mandi, cuci, kakus yang digunakan oleh rumah tangga siswa pendaftar beasiswa Bidikmisi.

13. Jumlah Tanggungan

Jumlah tanggungan adalah Jumlah yang ditanggung orang tua / wali saudara yang masih berusia dibawah 21 tahun dan diatas 21 tahun jika masih kuliah.

### 3.2 Struktur Data

Struktur data dari penelitian ini berdasarkan variabel-variabel yang telah disebutkan sebelumnya disajikan dalam Tabel 3.2.

**Tabel 3.2** Struktur Data

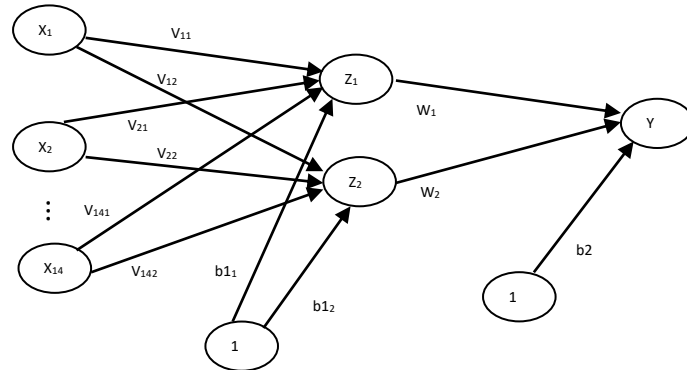
Nama Siswa	Nama Sekolah	Prediktor				Respon (Y)
		$X_1$	$X_2$	...	$X_{12}$	
1	1	$x_{11}$	$x_{21}$	...	$x_{121}$	$y_1$
2	2	$x_{12}$	$x_{22}$	...	$x_{122}$	$y_2$
3	3	$x_{13}$	$x_{23}$	...	$x_{123}$	$y_3$
4	4	$x_{14}$	$x_{24}$	...	$x_{124}$	$y_4$
5	5	$x_{15}$	$x_{25}$	...	$x_{125}$	$y_5$
⋮	⋮	⋮		⋮	⋮	⋮
52420	52420	$x_{152420}$	$x_{252420}$	...	$x_{252420}$	$y_{52420}$

### 3.3 Prosedur Penelitian

Metode dan tahapan Penelitian yang akan dilakukan untuk mencapai tujuan penelitian adalah sebagai berikut.

1. Melakukan *preprocessing* data Bidikmisi Tahun 2017 yaitu dengan menganalisis *missing values*. Mengubah variabel prediktor X yang bersifat kategori dengan  $m$  kategori, variabel X yang bersekala ordinal diubah menjadi nilai berurutan tetapi dalam selang  $[0,1]$ , sedangkan variabel X yang bersekala nominal diubah ke dalam  $m-1$  *dummy* kemudian membagi data menjadi himpunan data *testing* dan data *training*, dengan prosedur *10-foldcross-validation*.
2. Melakukan klasifikasi beasiswa Bidikmisi menggunakan Jaringan Syaraf *Backpropagation*:
  - a. Membangun model jaringan *backpropagation*, yaitu dengan menentukan jumlah neuron pada *input layer*, *hidden layer*, dan *output layer*. *Input layer* yang digunakan sebanyak 33 neuron, *hidden layer* yang digunakan sebanyak 1 *hidden layer* dan 2 *hidden layer* serta jumlah neuron yang digunakan pada masing masing *hidden layer* digunakan *trial and error*. (Dalam penelitian ini menggunakan *software R* dengan *package* *neuralnet*).
  - b. Menentukan bobot awal pada *input layer* dan *hidden layer* untuk proses *training*. Pada penelitian ini menggunakan *default* bobot awal pada *input layer* dan *hidden layer* (*startweight*) pada *package* *neuralnet* yaitu diberikan nilai random yang berdistribusi normal.
  - c. Menetapkan fungsi aktivasi pada *hidden layer* dan pada *output layer*. Pada penelitian ini menggunakan aktivasi *logistic fuction* atau fungsi sigmoid, pada *package* *neuralnet* terdapat dua aktivasi yaitu “*logistic*” dan “*tanh*”. Karena output keluaran pada penelitian ini berupa *binary* yaitu 0 (tidak diterima) dan 1 (diterima) sehingga digunakan *logistic fuction* atau fungsi sigmoid.
  - d. Melakukan proses pembelajaran pada data *training* dengan metode pelatihan *backpropagation* untuk menghitung bobot dan bias pada prediksi status beasiswa Bidikmisi.

- e. Melakukan pengujian dengan menggunakan data *testing* berdasarkan arsitektur yang diperoleh dari data *training*.
  - f. Menghitung ketepatan klasifikasi pada data *training* dan data *testing*.
  - g. Menghitung rata-rata ketepatan klasifikasi.
3. Algoritma *Neural Network* menggunakan *Genetic Algorithm*:
- I. Proses GA dilakukan sebelum proses Jaringan Syaraf *Backpropagation*. Diharapkan GA dapat melakukan optimalisasi *initial weight* sebelum bobot tersebut digunakan dalam proses Jaringan Syaraf *Backpropagation*. Operator GA yang digunakan adalah sebagai berikut.
    - a. Skema Pengkodean dan Inisialisasi Populasi.  
 Sekema pengkodean bilangan real dengan memetakan bias dan bobot JST ke dalam kromosom GA. Gambar 3.1 merupakan Arsitektur JST dipetakan kedalam kromosom GA seperti Gambar 3.2.  
 Inisialisasi populasi dengan membangkitkan bilangan random sebanyak bobot dan bias JST yang dipetakan kedalam kromosom.
    - b. Mengevaluasi nilai setiap masing-masing kromosom dengan nilai fitness. Dimana nilai fitness yang digunakan adalah nilai AUC.
    - c. Melakukan proses seleksi sebanyak N kromosom dari sejumlah P induk yang berasal dari populasi dengan seleksi *roulette wheel*. Kromosom dengan nilai fitness yang tinggi memiliki peluang yang lebih besar untuk terseleksi dalam memilih pasangan induk secara acak untuk bereproduksi.
    - d. Melakukan proses pindah silang.
    - e. Menghitung proses mutasi.
    - f. Pergantian populasi yang lama dengan populasi generasi yang baru dengan cara memilih kromosom terbaik dari induk dan anak baru yang memiliki nilai fitness tertinggi setelah terjadinya seleksi, pindah silang dan mutasi.
    - g. Melihat apakah solusi yang didapatkan sudah memenuhi kriteria atau belum. Apabila solusi yang didapatkan belum mencapai kriteria maka kembali kelangkah (b). Kriterianya adalah ketika nilai fitness terbaik sudah konvergen dari hasil generasi sebelumnya.



**Gambar3.1** Arsitektur JST

$V_{11}$	$V_{21}$	...	$V_{141}$	$V_{12}$	$V_{22}$	...	$V_{142}$	$W_1$	$W_2$	$b_{11}$	$b_{12}$	$b_2$
----------	----------	-----	-----------	----------	----------	-----	-----------	-------	-------	----------	----------	-------

**Gambar 3.2** Pengkodean Kromosom

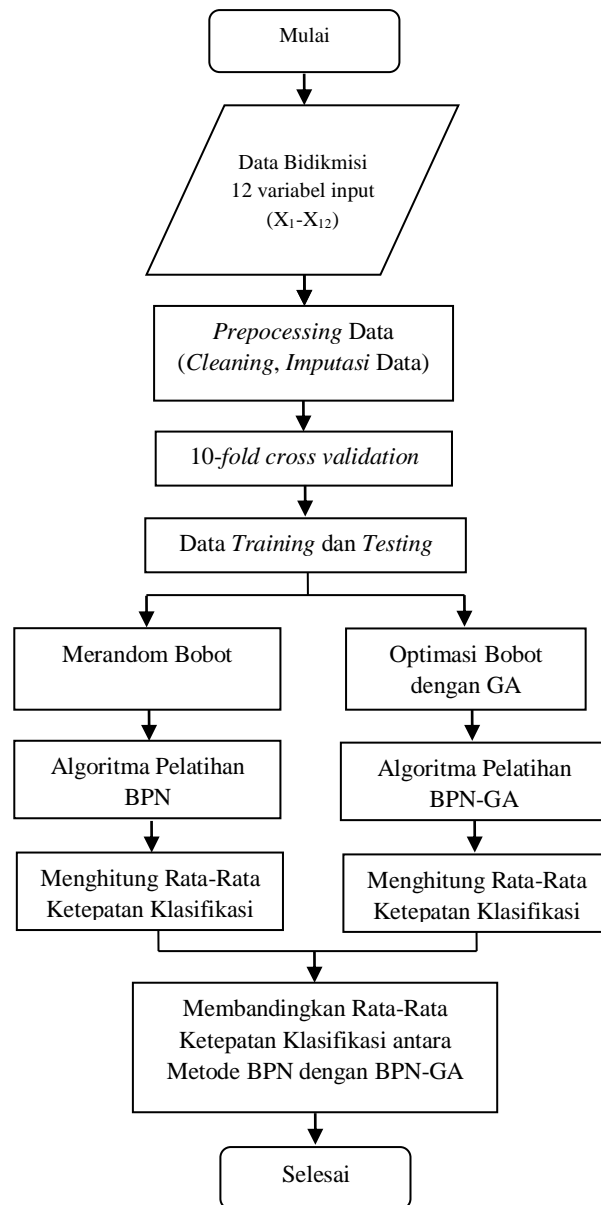
II. Menerapkan hasil bobot optimal oleh GA ke algoritma *Backpropagation*.

4. Membandingkan dan memilih metode terbaik untuk klasifikasi Bidikmisi di Jawa Timur. Metode terbaik adalah metode yang menghasilkan model yang mampu memberikan rata-rata ketepatan klasifikasi yang tinggi pada data *training* dan data *testing*.

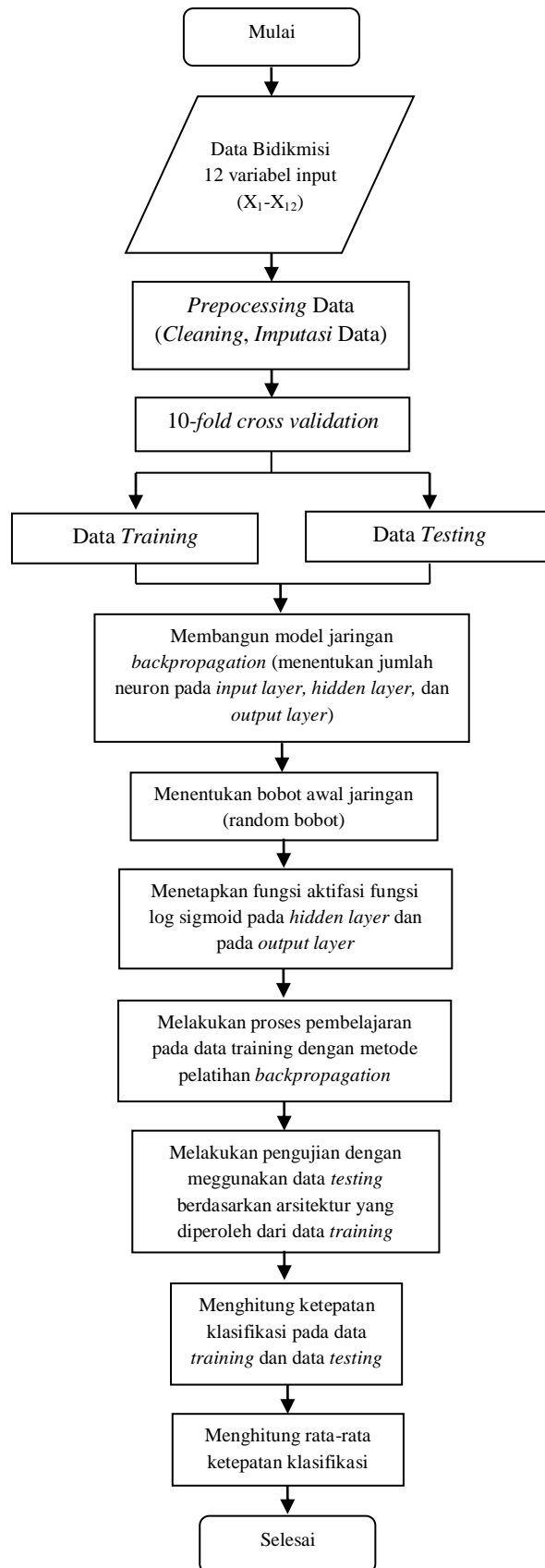
Langkah-langkah yang telah diuraikan dapat digambarkan secara umum menggunakan diagram yang terdapat pada Gambar 3.3 dan secara terperinci untuk masing-masing metode yaitu

1. Diagram alir Algoritma *Backpropagation* pada Gambar 3.4
2. Diagram alir Optimasi Bobot BPN dengan GA pada Gambar 3.5

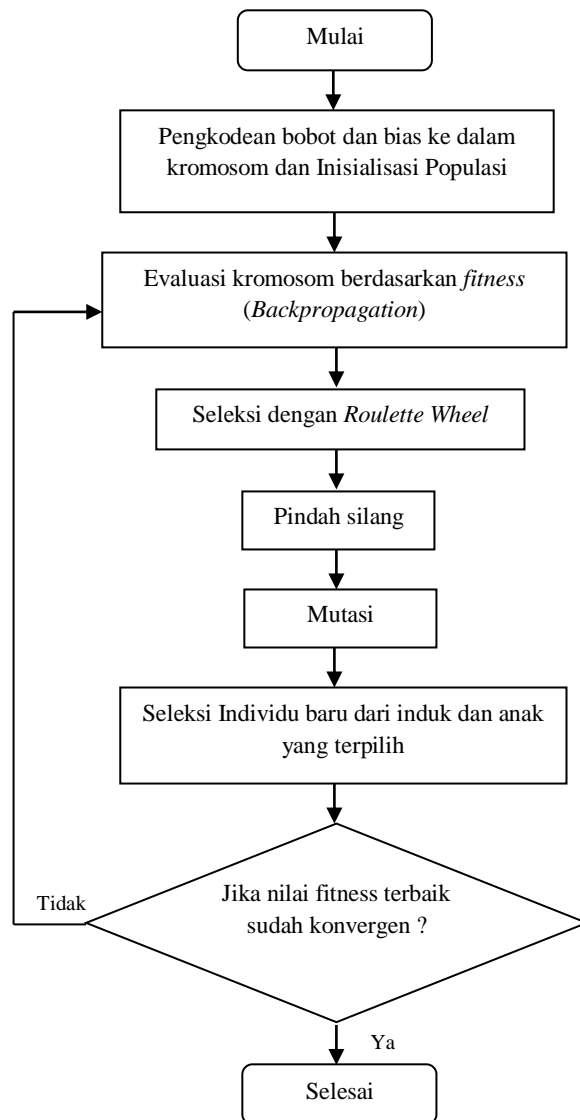




**Gambar 3.3** Diagram Alir Keseluruhan Penelitian



**Gambar 3.4** Diagram Alir *Backpropagation*



**Gambar 3.5** Diagram Alir Optimasi Bobot *Backpropagation* dengan GA

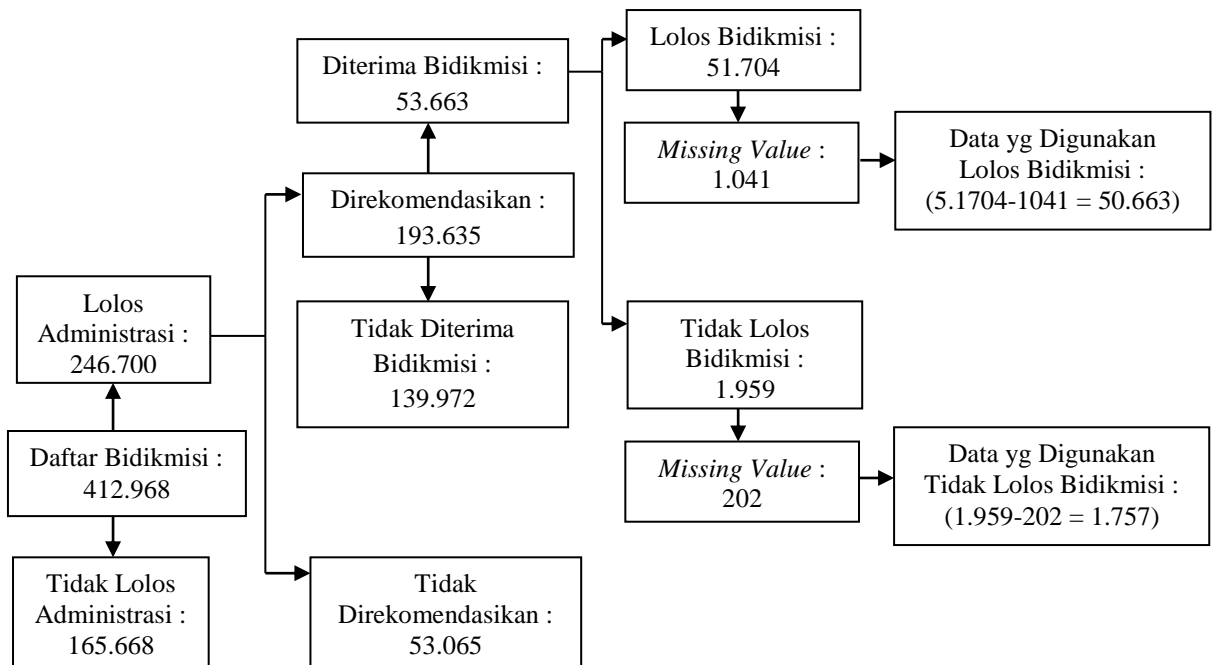
## BAB 4

### ANALISIS DAN PEMBAHASAN

Pada bab ini akan di uraikan klasifikasi penerimaan Beasiswa Bidikmisi di Provinsi Jawa Timur menggunakan *Neural Network Backpropagation* dengan satu *hidden layer* dan dua *hidden layer* dan optimasi bobot dan bias *Backpropagation* dengan *Genetic Algorithm* serta mengukur kinerja metode klasifikasi untuk menentukan metode terbaik untuk kasus Beasiswa Bidikmisi di Provinsi Jawa Timur.

#### 4.1 Alur Proses Data

Beasiswa Bidikmisi 2017 diterapkan diseluruh wilayah Indonesia yang meliputi 33 provinsi, 497 kabupaten/kota. Jawa Timur merupakan Pendaftar terbesar atau terbanyak. Sehingga dalam penelitian ini akan dibahas klasifikasi status penerimaan Beasiswa Bidikmisi di Jawa Timur.



**Gambar 4.1** Alur Data Proses Seleksi Pendaftaran Status Penerimaan Beasiswa Bidikmisi Di Jawa Timur

Gambar 4.1 disajikan alur data proses seleksi pendaftaran status penerimaan Beasiswa Bidikmisi di Jawa Timur. Diketahui jumlah pendaftar Beasiswa Bidikmisi di Jawa Timur Tahun 2017 sebanyak 412.968 siswa. Dari jumlah pendaftar Bidikmisi tersebut, sebanyak 246.700 siswa yang lolos administrasi dan 165.668 siswa yang tidak lolos administrasi. Dari jumlah siswa yang lolos administrasi, sebanyak 193.635 siswa yang lolos tahap perekomendasi sekolah atau siswa yang oleh sekolah dianggap memenuhi kriteria dan persyaratan dalam mendaftar Beasiswa bidikmisi, dan yang tidak lolos tahap perekomendasi sekolah sebanyak 53.065 siswa. Dari jumlah siswa yang lolos tahap perekomendasi sekolah tersebut, sebanyak 53663 siswa lolos seleksi berkas dari Bidikmisi Kemendikbud dan yang tidak lolos seleksi berkas sebanyak 139.972 siswa. Dari jumlah siswa yang lolos seleksi berkas, sebanyak 51.704 siswa yang diterima Beasiswa bidikmisi atau lolos Bidikmisi dan sebanyak 1.959 siswa yang tidak diterima Beasiswa Bidikmisi atau tidak lolos Bidikmisi.

Berdasarkan Gambar 4.1 diketahui jumlah siswa yang lolos Bidikmisi sebanyak 51.704 terdapat *missing value* sebanyak 1.041 sehingga data yang digunakan dalam penelitian ini untuk status lolos Bidikmisi atau diterima Beasiswa Bidikmisi sebanyak 50.633 ( $51.704 - 1.041 = 50.633$ ), sedangkan jumlah siswa yang tidak lolos Bidikmisi sebanyak 1.959 terdapat *missing value* sebanyak 202 sehingga data yang digunakan dalam penelitian ini untuk status tidak lolos Bidikmisi atau tidak diterima Beasiswa Bidikmisi sebanyak 1757 ( $1.959 - 202 = 1.757$ ). Tabel 4.1 disajikan proporsi status penerimaan Beasiswa Bidikmi.

**Tabel 4.1** Tabel Proposi Status Penerimaan Beasiswa Bidikmisi

Tidak Diterima Beasiswa Bidikmisi	Diterima Beasiswa Bidikmisi
3,35%	96,65%

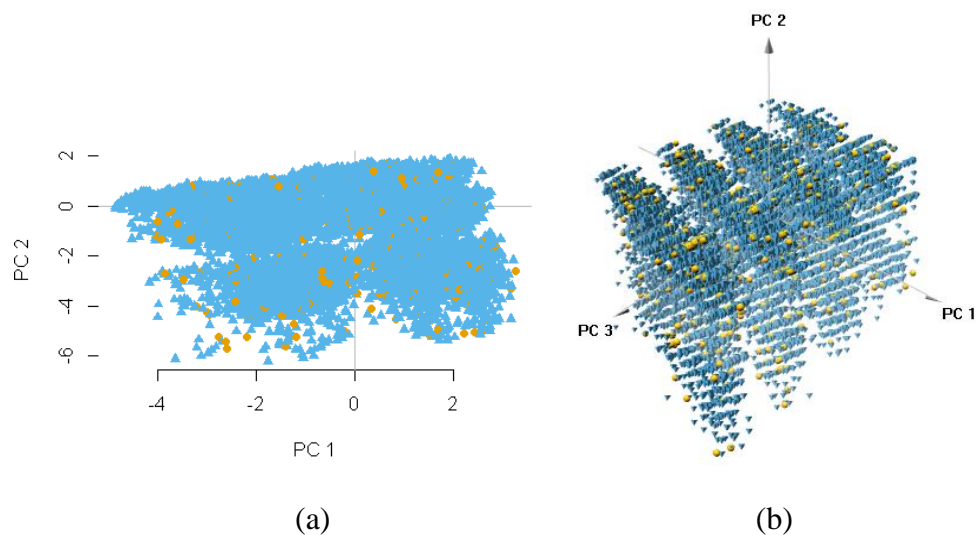
Proporsi tidak diterima Beasiswa Bidikmis:

$$\left( \frac{1.757}{50.633 + 1.757} \times 100 = 3,35\% \right)$$

Proporsi diterima Beasiswa Bidikmisi:

$$\left( \frac{50.633}{50.633 + 1.757} \times 100 = 96,65\% \right)$$

Tabel 4.1 diketahui bahwa proporsi diterima Bidikmisi sebesar 96,65% sedangkan proporsi tidak diterima Bidikmisi sebesar 3,35%, hal ini menunjukkan bahwa status diterima lebih banyak (mayoritas) dibandingkan dengan status tidak diterima lebih sedikit (minoritas) atau yang disebut dengan ketidak seimbangan kelas. Ketidak seimbangan kelas ini juga diketahui dengan melihat nilai dari IR (*Imbalance Ratio*) yaitu perbandingan antara prosentase kelas mayoritas dengan prosentase kelas minoritas yaitu sebesar 28,85%. Semakin tinggi nilai IR maka dataset semakin tidak seimbang.



**Gambar 4.2** (a) Gambar Ruang 2 Dimensi dan (b) Gambar Ruang 3 Dimensi Sebaran Data Status Beasiswa Bidikmisi Jawa Timur 2017

Selanjutnya untuk mengetahui pola sebaran kelas status Beasiswa Bidikmisi digunakan teknik PCA (*Principle Component Analysis*). PCA merupakan teknik untuk membangun variabel-variabel baru atau menyederhanakan suatu data yang merupakan kombinasi linier dari variabel-variabel asli. Jumlah variabel-variabel baru yang dihasilkan akan sama dengan

jumlah dari variabel-variabel lama, dan variabel-variabel baru ini tidak saling berkorelasi satu sama lain (Faisal, 2016). Gambar 4.2 (a) gambar ruang 2 dimensi dan Gambar 4.2 (b) gambar ruang 3 dimensi sebaran kelas status Beasiswa Bidikmisi Jawa Timur Tahun 2017. Gambar 4.2 (a) dan Gambar 4.2 (b) menunjukkan terjadi *overlapping* antar kedua kelas tersebut.

## 4.2 Analisis Karakteristik Data

Pengukuran penerimaan Beasiswa Bidikmisi dilakukan dengan pendekatan Data mikro yaitu karakteristik siswa pendaftar Beasiswa Bidikmisi di seluruh Kabupaten/Kota di Provinsi Jawa Timur. Terdapat 12 (dua belas) variabel yang mencerminkan karakteristik pendaftaran Beasiswa Bidikmisi. Berikut disajikan deskriptif karakteristik pendaftaran Beasiswa Bidikmisi terhadap katagori status penerimaan Beasiswa Bidikmisi.

**Tabel 4.2** Karakteristik Berdasarkan Status Penerimaan Beasiswa Bidikmisi dan Jenis Pekerjaan Ayah

Class Bidikmisi	Pekerjaan Ayah						Total
	Tidak Bekerja	Petani, Nelayan, Lainnya	TNI/Polri	Wirausaha	Pegawai Swasta	PNS	
Tidak Di Terima	197 0,38%	1024 1,95%	2 0,00%	245 0,47%	252 0,48%	37 0,07%	1757 3,35%
Di Terima	4882 9,31%	31613 60,31%	83 0,16%	6141 11,71%	7163 13,66%	783 1,49%	50663 96,65%
Total	5079 9,69%	32637 62,26%	85 0,16%	6386 12,18%	7413 14,14%	820 1,56%	52420

Pada Tabel 4.2 menyajikan hasil *cross tabulation* variabel jenis pekerjaan ayah ( $X_1$ ) dengan kelas katagori Status Penerimaan Beasiswa Bidikmisi ( $Y$ ). Berdasarkan Tabel 4.2 diketahui prosentase terbesar pekerjaan Ayah dari status penerimaan Beasiswa Bidikmisi dengan status tidak diterima yaitu sebagai Petani, Nelayan, dan lainnya sebesar 1,95% atau 1024 siswa sedangkan prosentase terkecil pekerjaan Ayah dari status penerimaan Beasiswa Bidikmisi dengan status tidak diterima yaitu sebagai TNI/Polri sebesar 0,00% atau 2 siswa. Prosentase

terbesar pekerjaan Ayah dari status penerimaan Beasiswa Bidikmisi dengan status diterima yaitu Petani, Nelayan dan Lainnya yaitu sebesar 60,31% atau 31613 sedangkan prosentase terkecil pekerjaan Ayah dari status penerimaan Beasiswa Bidikmisi dengan status diterima yaitu TNI/Polri sebesar 0,16% atau 83 siswa.

**Tabel 4.3** Karakteristik Berdasarkan Status Penerimaan Beasiswa Bidikmisi dan Jenis Pekerjaan Ibu

Class Bidikmisi	Pekerjaan Ibu						Total
	Tidak Bekerja	Petani, Nelayan, Lainnya	TNI/Polri	Wirausaha	Pegawai Swasta	PNS	
Tidak Di Terima	871 1,66%	610 1,16%	0 0,00%	158 0,30%	101 0,19%	17 0,03%	1757 3,35%
Di Terima	24144 46,06%	19.598 37,39%	2 0,00%	4287 8,18%	2417 4,61%	215 0,41%	50663 96,65%
Total	25015 47,72%	20208 38,55%	2 0,00%	2225 8,48%	2518 4,80%	232 0,44%	52420

Pada Tabel 4.3 menyajikan hasil *cross tabulation* variabel jenis pekerjaan Ibu (X<sub>2</sub>) dengan kelas katagori Status Penerimaan Beasiswa Bidikmisi (Y). Berdasarkan Tabel 4.3 diketahui prosentase terbesar pekerjaan Ibu dari status penerimaan Beasiswa Bidikmisi dengan status tidak diterima yaitu tidak bekerja sebesar 1,66% atau 871 siswa sedangkan prosentase terkecil pekerjaan Ibu dari status penerimaan Beasiswa Bidikmisi dengan status tidak diterima yaitu sebagai TNI/Polri sebesar 0,00% atau 0 siswa. Prosentase terbesar pekerjaan Ibu dari status penerimaan Beasiswa Bidikmisi dengan status diterima yaitu tidak bekerja sebesar 46,01% atau 24144 siswa sedangkan prosentase terkecil pekerjaan Ibu dari status penerimaan Beasiswa Bidikmisi dengan status diterima yaitu TNI/Polri sebesar 0,00% atau 2 siswa.



**Tabel 4.4** Karakteristik Berdasarkan Status Penerimaan Beasiswa Bidikmisi dan Pendidikan Ayah

Class Bidikmisi	Pendidikan Ayah				
	Tidak Sekolah	Pendidikan Dasar	SMA/MA	PT (D1,D2,D3,D4/S1)	PT (S2,S3)
Tidak Di Terima	54 0,1%	917 1,75%	634 1,21%	143 0,27%	9 0,02
Di Terima	1610 3,07%	26095 49,78%	19028 36,3%	3832 7,31%	98 0,19%
Total	1664 3,17%	27012 51,53%	19662 37,51%	3975 7,58%	107 0,2%

Pada Tabel 4.4 menyajikan hasil *cross tabulation* variabel jenis Pendidikan Ayah ( $X_3$ ) dengan kelas katagori Status Penerimaan Beasiswa Bidikmisi ( $Y$ ). Berdasarkan Tabel 4.4 diketahui prosentase terbesar Pendidikan Ayah dari status penerimaan Beasiswa Bidikmisi dengan status tidak diterima yaitu Pendidikan Dasar (SD/MI/ dan SMP/MTs) sebesar 1,75% atau 917 siswa sedangkan prosentase terkecil Pendidikan Ayah dari status penerimaan Beasiswa Bidikmisi dengan status tidak diterima yaitu tidak sekolah sebesar 0,1% atau 54 siswa. Prosentase terbesar Pendidikan Ayah dari status penerimaan Beasiswa Bidikmisi dengan status diterima yaitu Pendidikan Dasar (SD/MI/ dan SMP/MTs) sebesar 49,78% atau 26095 siswa sedangkan prosentase terkecil Pendidikan Ayah dari status penerimaan Beasiswa Bidikmisi dengan status diterima yaitu tidak sekolah sebesar 3,07% atau 1610 siswa.

Pada Tabel 4.5 menyajikan hasil *cross tabulation* variabel jenis Pendidikan Ibu ( $X_4$ ) dengan kelas katagori Status Penerimaan Beasiswa Bidikmisi ( $Y$ ). Berdasarkan Tabel 4.5 diketahui prosentase terbesar Pendidikan Ibu dari status penerimaan Beasiswa Bidikmisi dengan status tidak diterima Beasiswa Bidikmisi yaitu Pendidikan Dasar (SD/MI/ dan SMP/MTs) sebesar 1,99% atau 1043 siswa sedangkan prosentase terkecil Pendidikan Ibu dari status penerimaan Beasiswa Bidikmisi dengan status tidak diterima yaitu tidak sekolah sebesar 0,1% atau 54 siswa. Prosentase terbesar Pendidikan Ibu dari status penerimaan Beasiswa Bidikmisi dengan status diterima yaitu Pendidikan Dasar (SD/MI/ dan

SMP/MTs) sebesar 57,13% atau 29949 siswa sedangkan prosentase terkecil Pendidikan Ibu dari status penerimaan Beasiswa Bidikmisi dengan status diterima yaitu tidak sekolah sebesar 3,46% atau 1812 siswa.

**Tabel 4.5** Karakteristik Berdasarkan Status Penerimaan Beasiswa Bidikmisi dan Pendidikan Ibu

Class Bidikmisi	Pendidikan Ibu				
	Tidak Sekolah	Pendidikan	SMA/MA	PT (D1,D2,D3,D4/S1)	PT (S2,S3)
Tidak Di Terima	61 0,12%	1043 1,99%	523 1,00%	128 0,24%	2 0,00%
Di Terima	1812 3,46%	29949 57,13%	15724 30,00%	3148 6,01%	30 0,06%
Total	1873 3,57%	30992 59,12%	16247 30,99%	3276 6,25%	32 0,06%

**Tabel 4.6** Karakteristik Berdasarkan Status Penerimaan Beasiswa Bidikmisi dan Penghasilan Ayah

Class Bidikmisi	Penghasilan Ayah				
	1	2	3	4	5
Tidak Di Terima	268 0,51%	918 1,75%	424 0,81%	116 0,22%	31 0,06%
Di Terima	6983 13,32%	28556 54,48%	12068 23,02%	2632 5,02%	424 0,81%
Total	7251 13,83%	29474 56,23%	12492 23,83%	2748 5,24%	455 0,87%

Pada Tabel 4.6 menyajikan hasil *cross tabulation* variabel jenis Penghasilan Ayah ( $X_5$ ) dengan kelas katagori Status Penerimaan Beasiswa Bidikmisi ( $Y$ ). Berdasarkan Tabel 4.6 diketahui prosentase terbesar Penghasilan Ayah dari status penerimaan Beasiswa Bidikmisi dengan status tidak diterima yaitu dengan katagori 2 (kurang dari Rp. 1000.000) sebesar 1,75% atau 918 siswa sedangkan prosentase terkecil Penghasilan Ayah dari status penerimaan Beasiswa Bidikmisi dengan status tidak diterima yaitu dengan katagori 5 (diatas Rp. 3000.001) sebesar 0,06% atau 31 siswa. Prosentase terbesar Penghasilan Ayah dari status penerimaan Beasiswa Bidikmisi dengan status diterima yaitu dengan

katagori 2 (kurang dari Rp. 1000.000) sebesar 54,48% atau 28556 siswa sedangkan prosentase terkecil Penghasilan Ayah dari status penerimaan Beasiswa Bidikmisi dengan status diterima yaitu dengan katagori 5 (diatas Rp. 3000.001) sebesar 0,81% atau 424 siswa.

**Tabel 4.7** Karakteristik Berdasarkan Status Penerimaan Beasiswa Bidikmisi dan Penghasilan Ibu

Class Bidikmisi	Penghasilan Ibu				
	1	2	3	4	5
Tidak Di Terima	950 1,81%	662 1,26%	106 0,20%	29 0,06%	10 0,02%
Di Terima	27143 51,78%	20435 38,98%	2546 4,86%	440 0,84%	99 0,19%
Total	28093 53,59%	21097 40,25%	2652 5,06%	469 0,89%	109 0,21%

Pada Tabel 4.7 menyajikan hasil *cross tabulation* variabel jenis Penghasilan Ibu ( $X_6$ ) dengan kelas katagori Status Penerimaan Beasiswa Bidikmisi (Y). Berdasarkan Tabel 4.7 diketahui prosentase terbesar Penghasilan Ibu dari status penerimaan Beasiswa Bidikmisi dengan status tidak diterima yaitu dengan katagori 1 (tidak berpenghasilan) sebesar 1,81% atau 950 siswa sedangkan prosentase terkecil Penghasilan Ibu dari status penerimaan Beasiswa Bidikmisi dengan status tidak diterima yaitu dengan katagori 5 (diatas Rp. 3000.001) sebesar 0,02% atau 10 siswa. Prosentase terbesar Penghasilan Ibu dari status penerimaan Beasiswa Bidikmisi dengan status diterima yaitu dengan katagori 1 (tidak berpenghasilan) sebesar 51,78% atau 27143 siswa sedangkan prosentase terkecil Penghasilan Ibu dari status penerimaan Beasiswa Bidikmisi dengan status diterima yaitu dengan katagori 5 (diatas Rp. 3000.001) sebesar 0,21% atau 109 siswa.

**Tabel 4.8** Karakteristik Berdasarkan Status Penerimaan Beasiswa Bidikmisi dan Kepemilikan Rumah

Class Bidikmisi	Kepemilikan Rumah		
	Tidak Memiliki	Sewa	Sendiri
Tidak Di Terima	22 0,04%	507 0,97%	1228 2,34%
Di Terima	599 1,14%	15118 28,84%	34946 66,67%
Total	621 1,18%	15625 29,81%	36174 69,01%

Tabel 4.8 menyajikan hasil *cross tabulation* variabel Kepemilikan Rumah (X<sub>7</sub>) dengan kelas katagori status penerimaan Beasiswa Bidikmisi (Y). Berdasarkan Tabel 4.8 dapat diketahui bahwa prosentase terbesar kepemilikan rumah dari status penerimaan Beasiswa Bidikmisi dengan status tidak diterima yaitu milik sendiri sebesar 2,34% atau 1228 siswa sedangkan prosentase terkecil kepemilikan rumah dari status penerimaan Beasiswa Bidikmisi dengan status tidak diterima yaitu tidak memiliki rumah sebesar 0,04% atau 22 siswa. Prosentase terbesar kepemilikan rumah dari status penerimaan Beasiswa Bidikmisi dengan status diterima yaitu milik sendiri sebesar 66,67% atau 34946 siswa sedangkan prosentase terkecil kepemilikan rumah dari status penerimaan Beasiswa Bidikmisi dengan status diterima yaitu tidak memiliki rumah sebesar 1,14% atau 599 siswa.

Tabel 4.9 menyajikan hasil *cross tabulation* variabel Sumber listrik (X<sub>8</sub>) dengan kelas katagori status penerimaan Beasiswa Bidikmisi (Y). Berdasarkan Tabel 4.9 dapat diketahui bahwa prosentase terbesar sumber listrik dari status penerimaan Beasiswa Bidikmisi dengan status tidak diterima yaitu menggunakan PLN sebesar 3,32% atau 1741 siswa sedangkan prosentase terkecil sumber listrik dari status penerimaan Beasiswa Bidikmisi dengan status tidak diterima yaitu tidak ada listrik sebesar 0,01% atau 7 siswa. Prosentase terbesar sumber listrik dari status penerimaan Beasiswa Bidikmisi dengan status diterima yaitu menggunakan PLN sebesar 95,48% atau 50051 siswa sedangkan prosentase

terkecil sumber listrik dari status penerimaan Beasiswa Bidikmisi dengan status diterima yaitu tidak terdapat listrik sebesar 0,85% atau 444 siswa.

**Tabel 4.9** Karakteristik Berdasarkan Status Penerimaan Beasiswa Bidikmisi dan Sumber Listrik

Class Bidikmisi	Sumber Listrik		
	Tidak Ada	Genset, Tenaga Surya	PLN
Tidak Di Terima	7 0,01%	9 0,02%	1741 3,32%
Di Terima	444 0,85%	168 0,32%	50051 95,48%
Total	451 0,86%	177 0,34%	51792 98,80%

**Tabel 4.10** Karakteristik Berdasarkan Status Penerimaan Beasiswa Bidikmisi dan Luas Tanah

Class Bidikmisi	Luas Tanah		
	25-50 m <sup>2</sup>	50-99 m <sup>2</sup>	Lebih dari 100 m <sup>2</sup>
Tidak Di Terima	401 0,76%	340 0,65%	1016 1,94%
Di Terima	8895 16,97%	10727 20,46%	31041 59,22%
Total	9296 17,73%	11067 21,11%	32057 61,15%

Tabel 4.10 menyajikan hasil *cross tabulation* variabel Luas tanah (X<sub>9</sub>) dengan kelas katagori status penerimaan Beasiswa Bidikmisi (Y). Berdasarkan Tabel 4.10 dapat diketahui bahwa prosentase terbesar luas tanah dari status penerimaan Beasiswa Bidikmisi dengan status tidak diterima Beasiswa Bidikmisi yaitu lebih dari 100 m<sup>2</sup> sebesar 1,94% atau 1016 siswa sedangkan prosentase terkecil luas tanah dari status penerimaan Beasiswa Bidikmisi dengan status tidak diterima yaitu lebih dari 50-99 m<sup>2</sup> sebesar 0,65% atau 340 siswa. Prosentase

terbesar luas tanah dari status penerimaan Beasiswa Bidikmisi dengan status diterima yaitu lebih dari 100 m<sup>2</sup> sebesar 59,22% atau 31041 siswa sedangkan prosentase terkecil luas tanah dari status penerimaan Beasiswa Bidikmisi dengan status diterima yaitu lebih dari 25-50 m<sup>2</sup> sebesar 16,97% atau 8895 siswa.

Tabel 4.11 menyajikan hasil *cross tabulation* variabel Luas bangunan (X<sub>10</sub>) dengan kelas katagori status penerimaan Beasiswa Bidikmisi (Y). Berdasarkan Tabel 4.11 dapat diketahui bahwa prosentase terbesar luas bangunan dari status penerimaan Beasiswa Bidikmisi dengan status tidak diterima Beasiswa Bidikmisi yaitu 25-50 m<sup>2</sup> sebesar 1,47% atau 770 siswa sedangkan prosentase terkecil luas bangunan dari status penerimaan Beasiswa Bidikmisi dengan status tidak diterima yaitu lebih dari 100 m<sup>2</sup> sebesar 0,65% atau 342 siswa. Prosentase terbesar luas bangunan dari status penerimaan Beasiswa Bidikmisi dengan status diterima yaitu 50-99 m<sup>2</sup> sebesar 40,60% atau 21281 siswa sedangkan prosentase terkecil luas bangunan dari status penerimaan Beasiswa Bidikmisi dengan status diterima yaitu lebih dari 100 m<sup>2</sup> sebesar 15,76% atau 8264 siswa.

**Tabel 4.11** Karakteristik Berdasarkan Status Penerimaan Beasiswa Bidikmisi dan Luas Bangunan

Class Bidikmisi	Luas Bangunan		
	25-50 m <sup>2</sup>	50-99 m <sup>2</sup>	Lebih dari 100 m <sup>2</sup>
Tidak Di	770	645	342
Terima	1,47%	1,23%	0,65%
Di	21118	21281	8264
Terima	40,29%	40,60%	15,76%
Total	21888	21926	8606
	41,76%	41,83%	16,42%

Tabel 4.12 menyajikan hasil *cross tabulation* variabel Kepemilikan Fasilitas Mandi Cuci Kakus / MCK (X<sub>11</sub>) dengan kelas katagori status penerimaan Beasiswa Bidikmisi (Y). Berdasarkan Tabel 4.12 dapat diketahui bahwa prosentase terbesar MCK dari status penerimaan Beasiswa Bidikmisi dengan

status tidak diterima Beasiswa Bidikmisi yaitu memiliki MCK sendiri didalam sebesar 2,50% atau 1312 siswa sedangkan prosentase terkecil MCK dari status penerimaan Beasiswa Bidikmisi dengan status tidak diterima yaitu berbagi pakai sebesar 0,25% atau 133 siswa. Prosentase terbesar MCK dari status penerimaan Beasiswa Bidikmisi dengan status diterima yaitu memiliki MCK sendiri didalam sebesar 68,39% atau 35852 siswa sedangkan prosentase terkecil MCK dari status penerimaan Beasiswa Bidikmisi dengan status diterima yaitu berbagi pakai sebesar 9,01% atau 4724 siswa.

**Tabel 4.12** Karakteristik Berdasarkan Status Penerimaan Beasiswa Bidikmisi dan Kepemilikan MCK

Class Bidikmisi	Kepemilikan Fasilitas MCK		
	Kepemilikan sendiri didalam	Kepemilikan sendiri diluar	Berbagi Pakai
Tidak Di Terima	1312	312	133
	2,50%	0,60%	0,25%
Di Terima	35852	10087	4724
	68,39%	19,24%	9,01%
Total	37164	10.399	4.857
	70,90%	19,84%	9,27%

Tabel 4.13 menyajikan hasil *cross tabulation* variabel jumlah tanggungan ( $X_{12}$ ) dengan kelas katagori status penerimaan Beasiswa Bidikmisi (Y). Berdasarkan Tabel 4.13 dapat diketahui bahwa prosentase terbesar jumlah tanggungan dari status penerimaan Beasiswa Bidikmisi dengan status tidak diterima Beasiswa Bidikmisi yaitu 2 sebesar 1,19% atau 622 siswa sedangkan prosentase terkecil jumlah tanggungan dari status penerimaan Beasiswa Bidikmisi dengan status tidak diterima yaitu  $\leq 1$  sebesar 0,48% atau 252 siswa. Prosentase terbesar jumlah tanggungan dari status penerimaan Beasiswa Bidikmisi dengan status diterima yaitu 2 sebesar 34,86% atau 18273 siswa sedangkan prosentase terkecil jumlah tanggungan dari status penerimaan Beasiswa Bidikmisi dengan status diterima yaitu  $\leq 1$  sebesar 14,45% atau 7573 siswa.

**Tabel 4.13** Karakteristik Berdasarkan Status Penerimaan Beasiswa Bidikmisi dan Jumlah Tanggungan

Class Bidikmisi	Jumlah Tanggungan			
	$\leq 1$	2	3	$\geq 4$
Tidak Di Terima	252 0,48%	622 1,19%	448 0,85%	435 0,83%
Di Terima	7573 14,45%	18273 34,86%	12815 24,45%	12002 22,90%
Total	7825 14,93%	18895 36,05%	13263 25,30%	12437 23,73%

### 4.3 Prediksi *Neural Network Backpropagation*

Tujuan yang ingin dicapai dalam sub bab ini yaitu membangun suatu model prediksi *Neural Network Backpropagation* dalam memprediksi variabel respon penerimaan Beasiswa Bidikmisi di Jawa Timur. Sehingga untuk membangun suatu model *Neural Network* yang terdiri dari *input layer*, *hidden layer* dan *output layer*, maka akan ditentukan jumlah neuron pada masing-masing layer tersebut.

Jumlah neuron yang digunakan pada penelitian ini pada *input layer* yaitu terdiri dari 12 variabel yang merupakan karakteristik data Bidikmisi di Jawa Timur. Hasil *preprocessing* data ke dalam range  $[0,1]$  menghasilkan 23 neuron input untuk *Neural Network* yang akan dibangun. Penentuan jumlah *hidden layer* sangat penting untuk memaksimalkan kinerja *Neural Network* selama proses pelatihan. Sebelum menentukan jumlah neuron pada *hidden layer* dibentuk terlebih dahulu jumlah neuron pada *output layer*. Pada penelitian ini adalah klasifikasi biner dimana variabel respon atau targetnya terdiri dari 2 kategori yaitu tidak diterima (0) dan diterima (1).

Selanjutnya sebelum membangun suatu model *Neural Network Backpropagation* data dibagi menjadi data *training* dan data *testing* dengan 10 *cross fold validation* ( $k=10$ ).



#### 4.3.1 Model Prediksi *Neural Network Backpropagation 1 Hidden layer*

Berikut merupakan hasil klasifikasi dengan 1 *hidden layer*, untuk mendapatkan hasil klasifikasi yang baik harus menentukan jumlah *neuron* yang tepat pada *hidden layer* yaitu dilakukan *trial* dan *error* yaitu menggunakan jumlah neuron 2, 4, 6, 8 10, 12, 24 dengan menggunakan fungsi aktivasi *sigmoid biner* pada *hidden layer* dan *output layer*. Hasil yang diperoleh disajikan pada tabel 4.14. Untuk memperoleh hasil klasifikasi ini digunakan *software R* dengan *package* *neuralnet*.

**Tabel 4.14** Tingkat Kinerja AUC, G-Mean, dan Akurasi Data *Testing* dan Data *Training* dengan 1 *Hidden Layer*

Jumlah Neuron	Data <i>Testing</i>			Data <i>Training</i>		
	AUC	G-mean	Akurasi	AUC	G-mean	Akurasi
<b>4</b>	<b>0,51</b>	<b>0,01</b>	<b>78,00</b>	<b>0,51</b>	<b>0,00</b>	<b>78,02</b>
2	0,51	0,02	68,33	0,50	0,03	68,34
6	0,50	0,02	96,13	0,50	0,04	95,53
12	0,50	0,01	96,41	0,50	0,02	96,37
8	0,50	0,00	96,65	0,50	0,00	96,65
24	0,50	0,00	96,65	0,50	0,00	96,65

Tabel 4.14 disajikan tingkat kinerja AUC, G-Mean, dan Akurasi data *testing* dan data *training* dengan 1 *hidden layer*. Untuk mendapatkan *performance* yang terbaik dari hasil kinerja klasifikasi pada Tabel 4.15, yaitu dengan mengurutkan (*sort*) dari nilai tertinggi sampai terendah. Kriteria AUC yang paling utama digunakan dalam pemilihan model, karena data *imbalance* sehingga AUC lebih tepat untuk memilih model yang terbaik. Terdapat nilai G-mean 0,00 artinya semua kelas negatif tidak dapat diprediksi oleh model. Berdasarkan Tabel 4.14 dihasilkan ketepatan klasifikasi dengan *performance* terbaik pada jumlah neuron 4, sehingga dalam penelitian ini menggunakan jumlah neuron 4. Hasil klasifikasi untuk 10-fold *cross validation* dengan satu *hidden layer* dengan jumlah neuron 4 disajikan pada Tabel 4.15 untuk data *training* dan Tabel 4.17 untuk data *testing*.

**Tabel 4.15** Ketepatan Klasifikasi 10-Fold Cross Validation Neural Network Backpropagation Pada Data Training dengan 1 hidden Layer

		Model Classification		Total
		0	1	
Fold 1	0	1574 99,56%	7 0,44%	1581
	1	45409 99,59%	188 0,41%	45597
Fold 2	0	0 0%	1582 100%	1582
	1	0 0%	45596 100%	45596
Fold 3	0	0 0%	1581 100%	1581
	1	0 0%	45597 100%	45597
Fold 4	0	0 0%	1581 100%	1581
	1	0 0%	45596 100%	45596
Fold 5	0	0 0%	1581 100%	1581
	1	0 0%	45597 100%	45597
Fold 6	0	0 0%	1582 100%	1582
	1	0 0%	45597 100%	45597
Fold 7	0	0 0%	1581 100%	1581
	1	0 0%	45597 100%	45597

**Tabel 4.15** Lanjutan Ketepatan Klasifikasi 10-Fold Cross Validation Neural Network Backpropagation Pada Data Training dengan 1 hidden Layer

Actual Classification		Model Classification		Total
		0	1	
Fold 8	0	0	1581	1581
		0%	100%	
	1	3	45594	45597
		0,01%	100%	
Fold 9	0	1581	0	1581
		100%	0%	
	1	45596	0	45596
		100%	0%	
Fold 10	0	0	1582	1582
		0%	100%	
	1	15	45582	45597
		0,03%	100%	

Tabel 4.15 disajikan Ketepatan Klasifikasi 10-Fold Cross Validation Neural Network Backpropagation pada data training dengan jumlah hidden layer 1 dan 4 neuron. Diketahui bahwa pada Fold 1 dari 1581 status penerimaan Beasiswa Bidikmisi pada kategori 0 (tidak diterima Beasiswa Bidikmisi) terdapat 99,56% atau tidak ada status penerimaan Beasiswa Bidikmisi yang tepat diklasifikasikan masuk dalam katagori null oleh model yang dibentuk, 7 atau 0,44% status penerimaan Beasiswa Bidikmisi diklasifikasikan kedalam katagori 2 (dua). Untuk katagori 1 (satu), dari 45597 status penerimaan Beasiswa Bidikmisi terdapat 99,59% status penerimaan Beasiswa Bidikmisi yang diklasifikasikan masuk kedalam katagori 0 (null), 188 atau 0,41% status penerimaan Beasiswa Bidikmisi tepat diklasifikasikan masuk dalam katagori 1 (diterima Beasiswa Bidikmsi). Dan begitu seterusnya untuk fold 2 sampai dengan fold 10. Rata-rata hasil klasifikasi disajikan pada Tabel 4.16.

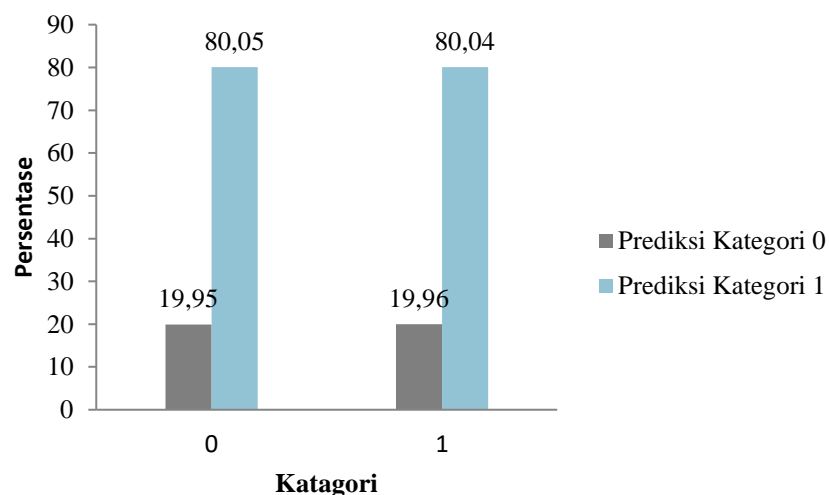
**Tabel 4.16** Rata-rata Ketepatan Klasifikasi Perkatagori Untuk 10-Fold Data Training Pada Model *Neural Network Backpropagation (1 Hidden Layer)*

Actual Classification	Model Classification		Akurasi
	Tidak diterima (0)	Diterima (1)	
Tidak diterima (0)	3155 19,95%	12658 80,05%	78,02%
Diterima (1)	91023 19,96%	364944 80,04%	

Sehingga dapat dihitung nilai akurasi atau ketepatan klasifikasi sebagai berikut.

$$\text{Akurasi} = \frac{364944 + 3155}{364944 + 91023 + 3155 + 12658} \times 100\% = 78,02\%$$

Tabel 4.16 disajikan Rata-rata ketepatan klasifikasi perkatagori untuk 10-Fold Data Training pada model *Neural Network Backpropagation* dengan 1 hidden layer. Diketahui hasil rata-rata ketepatan klasifikasi yang dibangun oleh model adalah sebesar 78,02%. Status penerimaan Beasiswa Bidikmisi yang berada pada katagori 0 (null) dapat tepat diklasifikasikan oleh model yaitu rata-rata sebesar 19,95%, sedangkan status penerimaan Beasiswa Bidikmisi yang berada pada katagori 1 (satu) tepat diklasifikasikan oleh model yaitu rata-rata sebesar 80,04%. Sebaran rata-rata klasifikasi yang dibangun oleh model *Neural Network Backpropagation* disajikan pada Gambar 4.3.



**Gambar 4.3** Sebaran Hasil Klasifikasi *Neural Network Backpropagation* dengan 1 Hidden layer 10-fold Data Training.

Sebaran rata-rata hasil klasifikasi *Neural Network Backpropagation* pada Gambar 4.3 menunjukkan bahwa pada katagori 0, model memprediksi atau mengklasifikasikan sebagian besar status Penerimaan Beasiswa Bidikmisi masuk pada katagori 1 (satu). Katagori 1, model memprediksi sebagian besar status penerimaan Beasiswa Bidikmisi masuk pada katagori 1 (satu).

**Tabel 4.17** Ketepatan Klasifikasi 10-Fold Cross Validation *Neural Network Backpropagation* Pada Data Testing dengan 1 Hidden Layer

	Actual Classification	Model Classification		Total
		0	1	
Fold 1	0	176	0	176
		100%	0%	
	1	5055	11	5066
		99,78%	0,22%	
Fold 2	0	0	175	175
		0%	100%	
	1	0	5067	5067
		0%	100%	
Fold 3	0	0	176	176
		0%	100%	
	1	0	5066	5066
		0%	100%	
Fold 4	0	0	176	176
		0%	100%	
	1	0	5067	5067
		0%	100%	
Fold 5	0	0	176	176
		0%	100%	
	1	0	5066	5066
		0%	100%	
Fold 6	0	0	175	175
		0%	100%	
	1	0	5066	5066
		0%	100%	

**Tabel 4.17** Lanjutan Ketepatan Klasifikasi 10-Fold Cross Validation Neural Network Backpropagation Pada Data Testing dengan 1 Hidden Layer

Actual Classification		Model Classification		Total
		0	1	
Fold 7	0	0	176	176
		0%	100%	
	1	0	5066	5066
		0%	100%	
Fold 8	0	0	176	176
		0%	100%	
	1	5	5061	5066
		0,10%	99,90%	
Fold 9	0	176	0	176
		100%	0%	
	1	5066	1	5067
		99,98%	0,02%	
Fold 10	0	0	175	175
		0%	100%	
	1	2	5064	5066
		0,04%	99,96%	

Tabel 4.17 disajikan Ketepatan Klasifikasi 10-Fold Cross Validation Neural Network Backpropagation pada data testing dengan jumlah hidden layer 1, dengan jumlah neuron 4. Diketahui bahwa pada Fold 8 dari 176 status penerimaan Beasiswa Bidikmisi pada kategori 0 (tidak diterima Beasiswa Bidikmisi) tidak terdapat status penerimaan Beasiswa Bidikmisi yang tepat diklasifikasikan masuk dalam katagori 0 (null) oleh model yang dibentuk, 176 atau 100% status penerimaan Beasiswa Bidikmisi diklasifikasikan kedalam katagori 1 (satu). Untuk katagori 1 (satu), dari 5066 status penerimaan Beasiswa Bidikmisi terdapat 5 atau 0,10% status penerimaan Beasiswa Bidikmisi yang diklasifikasikan masuk kedalam katagori 0 (null), 5061 atau 99,90% status penerimaan Beasiswa Bidikmisi tepat diklasifikasikan masuk dalam katagori 1 (diterima Beasiswa Bidikmisi). Dan begitu seterusnya untuk fold-fold yang lain. Rata-rata hasil klasifikasi disajikan pada Tabel 4.18.

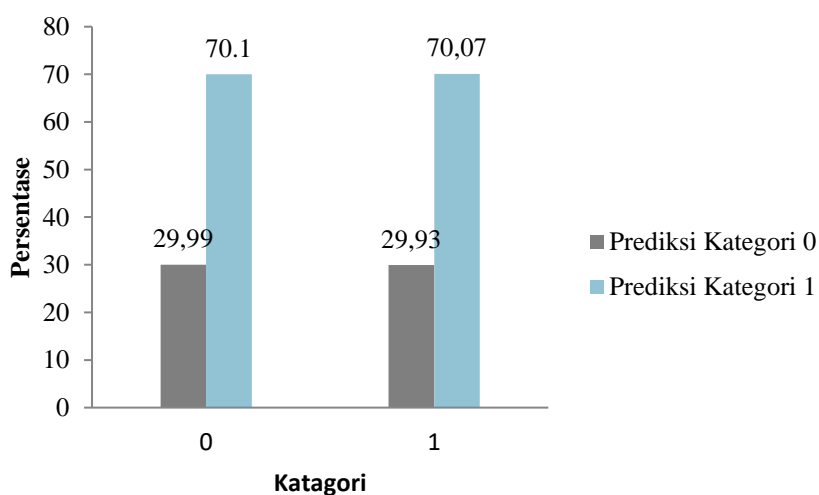
**Tabel 4.18** Rata-rata Ketepatan Klasifikasi Perkatagori Untuk 10-Fold Data Testing Pada Model *Neural Network Backpropagation (1 Hidden Layer)*

Actual Classification	Model Classification		Akurasi
	Tidak diterima (0)	Diterima (1)	
Tidak diterima (0)	352	1405	78,00%
	20,03%	79,97%	
Diterima (1)	10128	40535	78,00%
	19,99%	80,01%	

Sehingga dapat dihitung nilai akurasi atau ketepatan klasifikasi sebagai berikut.

$$\text{Akurasi} = \frac{352 + 40535}{40535 + 10128 + 352 + 1405} \times 100\% = 78,00\%$$

Tabel 4.18 disajikan Rata-rata ketepatan klasifikasi perkatagori untuk 10-Fold Data Testing pada model *Neural Network Backpropagation* dengan 1 *hidden layer*. Diketahui hasil rata-rata ketepatan klasifikasi yang dibangun oleh model adalah sebesar 78,00%. Status penerimaan Beasiswa Bidikmisi yang berada pada katagori 0 (null) dapat tepat diklasifikasikan oleh model yaitu rata-rata sebesar 20,03%, sedangkan status penerimaan Beasiswa Bidikmisi yang berada pada katagori 1 (satu) tepat diklasifikasikan oleh model yaitu rata-rata sebesar 80,01%. Sebaran rata-rata klasifikasi yang dibangun oleh model *Neural Network Backpropagation* disajikan pada Gambar 4.4.



**Gambar 4.4** Sebaran Hasil Klasifikasi *Neural Network Backpropagation* dengan 1 *Hidden layer* 10-fold Data Testing

Sebaran rata-rata hasil klasifikasi *Neural Network Backpropagation* pada Gambar 4.4 menunjukkan bahwa pada katagori 0, model memprediksi atau mengklasifikasikan sebgain besar status Penerimaan Beasiswa Bidikmisi masuk pada katagori 1 (satu). Katagori 1, model memprediksi sebagian besar status penerimaan Beasiswa Bidikmisi masuk pada katagori 1 (satu).

**Tabel 4.19** Hasil Kinerja Kasifikasi dengan Jumlah Neuron 4 Pada 1 *hidden layer*.

Fold	Data	AUC	G-mean	Akurasi %
1	<i>Training</i>	0,52	0,06	3,73
	<i>Testing</i>	0,56	0,05	3,57
2	<i>Training</i>	0,50	0,00	96,65
	<i>Testing</i>	0,50	0,00	96,66
3	<i>Training</i>	0,50	0,00	96,65
	<i>Testing</i>	0,50	0,00	96,64
4	<i>Training</i>	0,50	0,00	96,65
	<i>Testing</i>	0,50	0,00	96,64
5	<i>Training</i>	0,50	0,00	96,65
	<i>Testing</i>	0,50	0,00	96,64
6	<i>Training</i>	0,50	0,00	96,65
	<i>Testing</i>	0,50	0,00	96,66
7	<i>Training</i>	0,53	0,00	96,65
	<i>Testing</i>	0,50	0,00	96,64
8	<i>Training</i>	0,51	0,00	96,64
	<i>Testing</i>	0,51	0,00	96,55
9	<i>Training</i>	0,51	0,01	3,35
	<i>Testing</i>	0,49	0,01	3,38
10	<i>Training</i>	0,52	0,00	96,65
	<i>Testing</i>	0,50	0,00	96,62
Rata-Rata	<i>Training</i>	0,51	0,007	78,03
	<i>Testing</i>	0,51	0,006	78,00

Tabel 4.19 merupakan hasil kinerja klasifikasi dengan arsitektur yang paling baik yaitu dengan jumlah neuron 4 pada 1 *hidden layer*. Hasil analisis pada Tabel 4.19 menunjukkan bahwa ketepatan klasifikasi pada data *training* adalah 78,03% dan data *testing* sebesar 78,00%. Diketahui bahwa ketepatan klasifikasi dari model *Neural Network Backpropagation* yang terbentuk ditinjau dari akurasi



sebesar 78,00%, ditinjau dari AUC sebesar 0.51 dan ditinjau dari G-mean sebesar 0,006. Dalam hal ini hasil dari data *training* dan data *testing* menghasilkan selisih yang hampir sama menunjukkan model yang dibentuk cukup baik.

#### 4.3.2 Model Prediksi *Neural Network Backpropagation 2 Hidden Layer*

Seperti halnya penjelasan pada 1 *hidden layer*, pada 2 *hidden layer* juga digunakan *trial and error* untuk mendapatkan hasil prediksi yang baik yaitu dengan menggunakan *trial and error* dengan jumlah neuron 2-6, 4-2, 4-8, 5-9, 6-10, 10-4, 12-6, 12-8, 24-12, dan 24-30 dengan menggunakan fungsi aktivasi *sigmoid biner* pada *hidden layer* dan *output layer*.

**Tabel 4.20** Tingkat Kinerja AUC, G-Mean, dan Akurasi Data *Testing* dan Data *Training* dengan 2 *Hidden Layer*

Jumlah Neuron	Data <i>Testing</i>			Data <i>Training</i>		
	AUC	G-mean	Akurasi	AUC	G-mean	Akurasi
<b>10-4</b>	<b>0,51</b>	<b>0,01</b>	<b>68,73</b>	<b>0,50</b>	<b>0,01</b>	<b>68,72</b>
4-8	0,51	0,00	87,31	0,50	0,00	87,32
12-8	0,50	0,05	83,28	0,50	0,00	77,99
5-9	0,50	0,01	68,6	0,50	0,01	68,58
12-6	0,50	0,01	87,18	0,50	0,00	87,31
6-10	0,50	0,00	87,32	0,50	0,00	87,32
24-12	0,50	0,00	87,32	0,50	0,00	87,32
4-2	0,50	0,02	68,89	0,50	0,02	68,88
24-30	0,50	0,00	77,99	0,50	0,00	77,99
2-6	0,49	0,01	68,59	0,49	0,01	68,61

Berdasarkan Tabel 4.20 disajikan tingkat kinerja AUC, G-Mean, dan Akurasi data *testing* dan data *training* dengan 2 *hidden layer*. Untuk mendapatkan *performance* yang terbaik dari hasil kinerja klasifikasi pada Tabel 4.20, yaitu dengan mengurutkan (*sort*) dari nilai tertinggi sampai terendah. Kriteria AUC yang paling utama digunakan dalam pemilihan model, karena data *imbalance* sehingga AUC lebih tepat untuk memilih model yang terbaik. Terdapat nilai G-

mean 0,00 artinya semua kelas negatif tidak dapat diprediksi oleh model. Berdasarkan Tabel 4.20 dihasilkan ketepatan klasifikasi dengan *performance* terbaik pada jumlah neuron 10 pada *hidden layer* pertama dan jumlah neuron 4 pada *hidden layer* kedua, sehingga dalam penelitian ini menggunakan jumlah neuron 12 dan 4 untuk masing-masing *hidden*.

Hasil klasifikasi untuk 10-fold cross validation dengan 2 *hidden layer* dihasilkan *performance* yang terbaik dengan (10-4) jumlah neuron pada 2 *hidden layer* yang disajikan pada Tabel 4.23 untuk data *training* dan Tabel 4.25 untuk data *testing*. Tabel 4.23 merupakan klasifikasi dengan dua *hidden layer*, dimana menggunakan sigmoid biner pada 2 *hidden layer* dan output layer dengan jumlah 10 neuron pada *hidden layer* pertama dan 4 neuron pada *hidden layer* kedua.

**Tabel 4.21** Ketepatan Klasifikasi dengan 2 *Hidden layer* 10-Fold Cross Validation Neural Network Backpropagation Pada Data Training

Actual Classification		Model Classification		Total
		0	1	
Fold 1	0	1581	0	1581
		100.00%	0.00%	
	1	45596	1	45597
		100.00%	0.00%	
Fold 2	0	1576	6	1582
		100%	0%	
	1	45295	301	45596
		99%	1%	
Fold 3	0	0	1581	1581
		0%	100%	
	1	0	45597	45597
		0%	100%	
Fold 4	0	0	1581	1581
		0%	100%	
	1	0	45596	45596
		0%	100%	

**Tabel 4.21** Lanjutan Ketepatan Klasifikasi dengan 2 *Hidden layer* 10-Fold *Cross Validation Neural Network Backpropagation* Pada Data *Training*

	Actual Classification	Model Classification		Total
		0	1	
Fold 5	0	0	1581	1581
		0%	100%	
	1	0	45597	45597
		0%	100%	
Fold 6	0	0	1582	1582
		0%	100%	
	1	0	45597	45597
		0%	100%	
Fold 7	0	0	1581	1581
		0%	100%	
	1	0	45597	45597
		0%	100%	
Fold 8	0	1581	0	1581
		100%	0%	
	1	45596	1	45597
		100.00%	0%	
Fold 9	0	0	1581	1581
		0%	100%	
	1	0	45596	45596
		0%	100%	
Fold 10	0	0	1582	1582
		0%	100%	
	1	0	45597	45597
		0.00%	100%	

Tabel 4.21 disajikan Ketepatan Klasifikasi 10-Fold *Cross Validation Neural Network Backpropagation* pada data *training* dengan jumlah *hidden layer* 2, *hidden layer* pertama berjumlah 10 neuron, *hidden layer* kedua berjumlah 4 neuron. Diketahui bahwa pada Fold 2 dari 1582 status penerimaan Beasiswa Bidikmisi pada kategori 0 (tidak diterima Beasiswa Bidikmisi) terdapat 99,62% atau 1576 status penerimaan Beasiswa Bidikmisi yang tepat diklasifikasikan masuk dalam katagori 0 (null) oleh model yang dibentuk, 6 atau 0,38% status

penerimaan Beasiswa Bidikmisi diklasifikasikan kedalam katagori 1 (satu). Untuk katagori 1 (satu), dari 45596 status penerimaan Beasiswa Bidikmisi terdapat 45295 atau 99,34% status penerimaan Beasiswa Bidikmisi yang diklasifikasikan masuk kedalam katagori 0 (null), 301 atau 0,66% status penerimaan Beasiswa Bidikmisi tepat diklasifikasikan masuk dalam katagori 1 (diterima Beasiswa Bidikmisi). Dan begitu seterusnya untuk *fold-fold* yang lain. Rata-rata hasil klasifikasi disajikan pada Tabel 4.22.

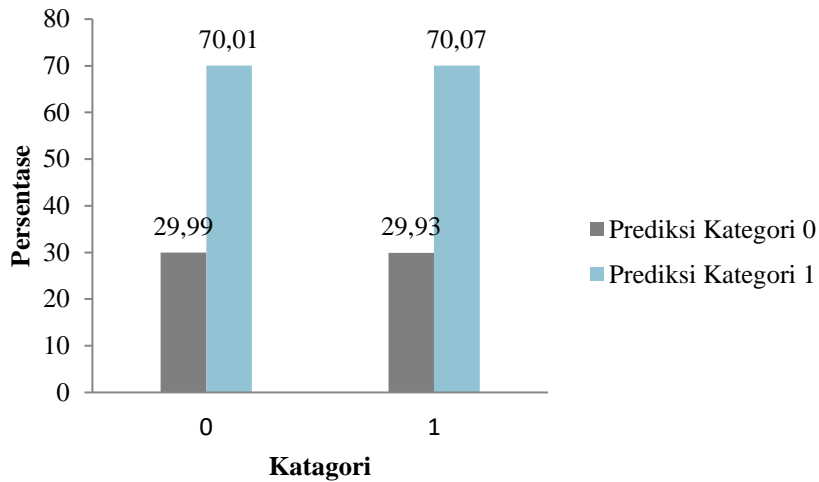
**Tabel 4.22** Rata-rata Ketepatan Klasifikasi Perkatagori Untuk 10-Fold Data Training Pada Model *Neural Network Backpropagation* (2 Hidden Liayer)

Actual Classification	Model Classification		Akurasi
	Tidak diterima (0)	Diterima (1)	
Tidak diterima (0)	4738	11075	68,72%
	29,96%	70,04%	
Diterima (1)	136487	319480	70,07%
	29,93%	70,07%	

Sehingga dapat dihitung nilai akurasi atau ketepatan klasifikasi sebagai berikut.

$$\text{Akurasi} = \frac{4738 + 319480}{319480 + 136487 + 4738 + 11075} \times 100\% = 68,72\%$$

Tabel 4.22 disajikan Rata-rata ketepatan klasifikasi perkatagori untuk 10-Fold Data Training pada model *Neural Network Backpropagation* pada 2 hidden layer. Diketahui hasil rata-rata ketepatan klasifikasi yang dibangun oleh model adalah sebesar 68,72%. Status penerimaan Beasiswa Bidikmisi yang berada pada katagori 0 (null) dapat tepat diklasifikasikan oleh model yaitu rata-rata sebesar 29,96%, sedangkan status penerimaan Beasiswa Bidikmisi yang berada pada katagori 1 (satu) tepat diklasifikasikan oleh model yaitu rata-rata sebesar 70,07%. Sebaran rata-rata klasifikasi yang dibangun oleh model *Neural Network Backpropagation* disajikan pada Gambar 4.5.



**Gambar 4.5** Sebaran Hasil Klasifikasi *Neural Network Backpropagation* dengan 2 *Hidden layer* 10-fold Data Training

Sebaran rata-rata hasil klasifikasi *Neural Network Backpropagation* pada Gambar 4.5 menunjukkan bahwa pada katagori 0, model memprediksi atau mengklasifikasikan sebagian besar status Penerimaan Beasiswa Bidikmisi masuk pada katagori 1 (satu). Katagori 1, model memprediksi sebagian besar status penerimaan Beasiswa Bidikmisi masuk pada katagori 1 (satu).

**Tabel 4.23** Ketepatan Klasifikasi denga 2 *Hidden layer* 10-Fold Cross Validation *Neural Network Backpropagation* Pada Data Testing

Actual Classification		Model Classification		Total
		0	1	
Fold 1	0	176	0	176
		100%	0%	
	1	5065	1	5066
		100%	0%	
Fold 2	0	175	0	175
		100%	0%	
	1	5034	33	5067
		99,35%	0,65%	

**Tabel 4.23** Lanjutan Ketepatan Klasifikasi denga 2 *Hidden layer* 10-Fold Cross Validation Neural Network Backpropagation Pada Data Testing

Actual Classification		Model Classification		Total
		0	1	
Fold 3	0	0	176	176
		0%	100%	
	1	0	5066	5066
		0%	100%	
Fold 4	0	0	176	176
		0%	100%	
	1	0	5067	5067
		0%	100%	
Fold 5	0	0	176	176
		0%	100%	
	1	0	5066	5066
		0%	100%	
Fold 6	0	0	175	175
		0%	100%	
	1	0	5066	5066
		0%	100%	
Fold 7	0	0	176	176
		0%	100%	
	1	0	5066	5066
		0%	100%	
Fold 8	0	176	0	176
		100%	0%	
	1	5065	1	5066
		100%	0%	
Fold 9	0	0	176	176
		0%	100%	
	1	0	5067	5067
		0%	100%	
Fold 10	0	0	175	175
		0%	100%	
	1	0	5066	5066
		0%	100%	

Tabel 4.23 disajikan Ketepatan Klasifikasi 10-Fold Cross Validation Neural Network Backpropagation pada data testing dengan jumlah hidden layer 2, hidden layer pertama berjumlah 10 neuron, hidden layer kedua berjumlah 4 neuron. Diketahui bahwa pada Fold 2 dari 175 status penerimaan Beasiswa Bidikmisi pada kategori 0 (tidak diterima Beasiswa Bidikmisi) terdapat 100% atau 175 status penerimaan Beasiswa Bidikmisi yang tepat diklasifikasikan masuk dalam katagori 0 (null) oleh model yang dibentuk, 0% status peneriman Beasiswa Bidikmisi diklasifikasikan kedalam katagori 1 (satu). Untuk katagori 1 (satu), dari 5067 status penerimaan Beasiswa Bidikmisi terdapat 5034 atau 99,35% status penerimaan Beasiswa Bidikmisi yang diklasifikasikan masuk kedalam katagori 0 (null), 33 atau 0,65% status penerimaan Beasiswa Bidikmisi tepat diklasifikasikan masuk dalam katagori 1 (diterima Beasiswa Bidikmsi). Dan begitu seterusnya untuk fold-fold yang lain. Rata-rata hasil klasifikasi disajikan pada Tabel 4.24.

**Tabel 4.24** Rata-rata Ketepatan Klasifikasi Perkatagori Untuk 10-Fold Data Testing Pada Model Neural Network Backpropagation (2 Hidden Layer)

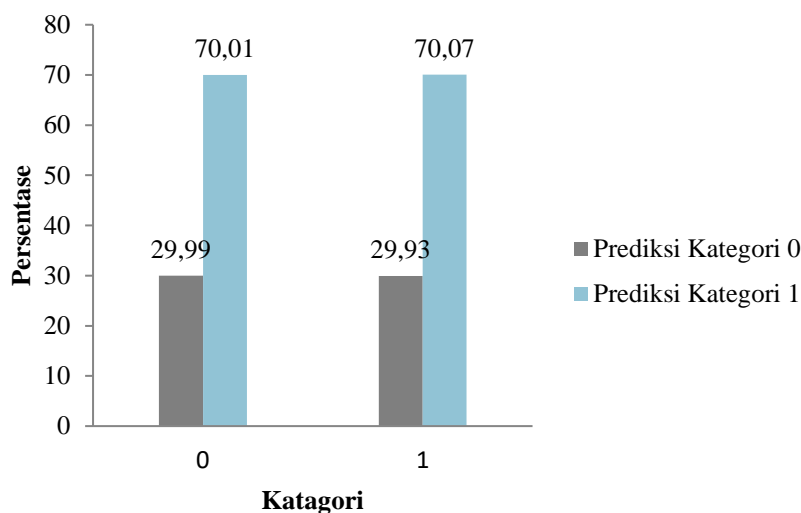
Actual Classification	Model Classification		Akurasi
	Tidak diterima (0)	Diterima (1)	
Tidak diterima (0)	527	1230	68,73%
	29,99%	70,01%	
Diterima (1)	15164	35499	68,73%
	29,93%	70,07%	

Sehingga dapat dihitung nilai akurasi atau ketepatan klasifikasi sebagai berikut.

$$\text{Akurasi} = \frac{527 + 35499}{35499 + 15164 + 527 + 1230} \times 100\% = 68,73\%$$

Tabel 4.24 disajikan Rata-rata ketepatan klasifikasi perkatagori untuk 10-Fold Data Testing pada model Neural Network Backpropagation dengan 2 hidden layer. Diketahui hasil rata-rata ketepatan klasifikasi yang dibangun oleh model adalah sebesar 68,73%. Status penerimaan Beasiswa Bidikmisi yang berada pada katagori 0 (null) dapat tepat diklasifikasikan oleh model yaitu rata-rata sebesar 29,99%, sedangkan status penerimaan Beasiswa Bidikmisi yang berada pada katagori 1 (satu) tepat diklasifikasikan oleh model yaitu rata-rata

sebesar 70,07%. Sebaran rata-rata klasifikasi yang dibangun oleh model *Neural Network Backpropagation* disajikan pada Gambar 4.7.



**Gambar 4.6** Sebaran Hasil Klasifikasi *Neural Network Backpropagation* dengan 2 *Hidden layer* 10-fold Data Testing.

Sebaran rata-rata hasil klasifikasi *Neural Network Backpropagation* pada Gambar 4.6 menunjukkan hasil yang hampir sama dengan sebaran rata-rata klasifikasi pada data *testing*, diketahui bahwa pada katagori 0, model memprediksi atau mengklasifikasikan sebagian besar status Penerimaan Beasiswa Bidikmisi masuk pada katagori 1 (satu). Katagori 1, model memprediksi sebagian besar status penerimaan Beasiswa Bidikmisi masuk pada katagori 1 (satu).

Hasil analisis pada Tabel 4.25 menunjukkan bahwa ketepatan klasifikasi pada data *training* adalah 68,72% dan data *testing* sebesar 68,73%. diketahui bahwa ketepatan klasifikasi dari model *Neural Network Backpropagation* yang terbentuk ditinjau dari akurasi sebesar 68,73%, ditinjau dari AUC sebesar 0,51 dan ditinjau dari G-mean sebesar 0,01. Dalam hal ini hasil dari data *training* dan data *testing* menghasilkan selisih yang hampir sama, sehingga dapat dikatakan model yang dibentuk cukup baik.



**Tabel 4.25** Hasil Kinerja Klasifikasi 10-Fold dengan Jumlah Neuron (10-4) dengan 2 Hidden Layer Pada Data Training Dan Testing

Fold	Data	AUC	G-mean	Akurasi %
1	Training	0,51	0,00	3,35
	Testing	0,51	0,01	3,38
2	Training	0,51	0,08	3,98
	Testing	0,56	0,08	3,97
3	Training	0,50	0,00	96,65
	Testing	0,50	0,00	96,64
4	Training	0,50	0,00	96,65
	Testing	0,50	0,00	96,64
5	Training	0,50	0,00	96,65
	Testing	0,50	0,00	96,64
6	Training	0,50	0,00	96,65
	Testing	0,50	0,00	96,66
7	Training	0,50	0,00	96,65
	Testing	0,50	0,00	96,64
8	Training	0,52	0,00	3,35
	Testing	0,52	0,01	3,38
9	Training	0,50	0,00	96,65
	Testing	0,50	0,00	96,64
10	Training	0,50	0,00	96,65
	Testing	0,50	0,00	96,66
Rata-Rata	Training	0,50	0,01	68,72
	Testing	0,51	0,01	68,73

#### 4.4 Optimasi Parameter Neural Network Backpropagation Menggunakan Algoritma Genetika (GA)

Neural Network (NN) dan Algoritma Genetika keduanya adalah alat yang kuat yang dimodelkan setelah fenomena alam. Neural Network dimodelkan seperti otak, dimana sangat paralel dan banyak menawarkan keuntungan ketika dalam pemecahan pengenalan pola dan masalah klasifikasi. Genetic Algorithm didasarkan pada teori evolusi dan kelangsungan hidup dan telah diterapkan untuk menyelesaikan banyak masalah pengoptimalan.

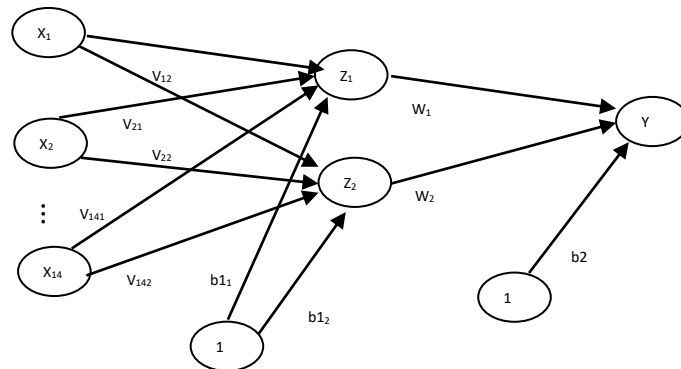
*Neural Network* memberikan banyak kelebihan dalam berbagai aplikasi, tetapi tidak efektif jika tidak dirancang dengan benar. Terdapat banyak pilihan dalam mendesain NN tetapi dengan memilih salah satu parameter yang tidak baik dapat menghasilkan hasil NN yang kurang baik. Untuk mendapatkan struktur parameter yang optimal pada *Neural Network Backpropagation* akan dilakukan optimasi menggunakan *Genetic Algorithm* (GA) untuk mendapatkan parameter yang optimum pada model *Neural Network Backpropagation*, sehingga didapatkan tingkat kinerja klasifikasi yang lebih baik.

Pada penelitian ini Genetika Algoritma digunakan untuk menemukan bobot dan bias awal yang optimal dari *Backpropagation*. Seperti yang disebutkan dalam Y.Du & Y.Li Tahun 2008, Algoritma Genetika baik digunakan untuk pengoptimalan global, sedangkan *Neural Network* baik digunakan dalam pengoptimalan lokal. Menggunakan kombinasi algoritma genetika untuk menentukan bobot dan bias awal dalam proses pelatihan *Backpropagation* untuk kesalahan lebih rendah dan untuk menghindari minimum lokal.

Seperti yang telah dijelaskan pada sub bab sebelumnya, langkah- langkah untuk mengoptimasi bobot dan bias awal pada *Backpropagation* hal pertama yang dilakukan adalah menentukan variabel-variabel yang dibutuhkan oleh Algoritma Genetika yaitu populasi, batas iterasi, peluang terjadinya pindah silang, peluang terjadinya mutasi dan kromosom untuk elitism. Dalam penelitian ini jumlah populasi yang digunakan sebanyak 50 kromosom dengan batas iterasi sebanyak 100, sebesar 0.8 peluang yang digunakan dalam pindah silang dan sebesar 0.1 peluang yang digunakan dalam mutasi. Kromosom yang terpilih sebanyak bobot yang digunakan sesuai dengan jumlah input, jumlah neuron pada jumlah *hidden layer* yang digunakan dan jumlah output.

Selanjutnya yang kedua yaitu inisialisasi kromosom, pada penelitian ini dilakukan optimasi parameter bobot *Neural Network* yaitu inisialisasi pengkodean dengan menggunakan *real valued* atau bilangan real. Proses yang dilakukannya yaitu dengan merepresentasikan bias dan bobot pada *Neural Network* kedalam bentuk kromosom GA. Gambar 4.7 (a). menyajikan contoh struktur *Neural Network* dengan 3 input dan 2 neuron pada *hidden layer*. Pembentukan kromosom yang direpresentasikan disajikan pada Gambar 4.7 (b).

(a)



(b).

$V_{11}$	$V_{21}$	...	$V_{141}$	$V_{12}$	$V_{22}$	...	$V_{142}$	$W_1$	$W_2$	$b_{11}$	$b_{12}$	$b_2$
----------	----------	-----	-----------	----------	----------	-----	-----------	-------	-------	----------	----------	-------


**Gambar 4.7** Pembentukan Kromosom Algoritma Genetika

Selanjutnya terbentuk 50 kromosom dengan menggeneret sejumlah populasi dengan nilai parameter yang bervariasi didalamnya. Nilai kromosom di generate diantara nilai range dari parameter (bias dan bobot).

Tahap ketiga fungsi fitness function yang digunakan adalah nilai dari AUC. Dari 50 kromosom yang telah degenerate kemudian dihitung nilai AUC dari tiap kromosomnya. Nilai AUC yang tinggi menghasilkan kromosom yang baik dan dapat bertahan hidup. Setelah mendapatkan kromosom yang baik kemudian tahap keempat menyeleksi dengan menggunakan *roulette wheel* yaitu tiap kromosom akan diseleksi untuk dijadikan sebagai calon orang tua. Pemilihan calon orang tua didasarkan pada nilai fitness yang telah didapatkan pada tahapan sebelumnya. Nilai frekuensi adalah fitness kumulatif yang dibagi dengan total fitness. Nilai tersebut akan menjadi batas terpilihnya kromosom. Kromosom yang terpilih jika nilai dari bilangan random terletak pada range nilai frekuensi kromosom sebelumnya dan kromosom tersebut.

Tahapan kelima yaitu proses pindah silang :

V <sub>11_1</sub>	V <sub>21_1</sub>	...	V <sub>141_1</sub>	V <sub>12_1</sub>	V <sub>22_1</sub>	...	V <sub>142_1</sub>	W <sub>1_1</sub>	W <sub>2_1</sub>	b <sub>1_1</sub>	b <sub>1_2_1</sub>	b <sub>2_1</sub>
V <sub>11_2</sub>	V <sub>21_2</sub>	...	V <sub>141_2</sub>	V <sub>12_2</sub>	V <sub>22_2</sub>	...	V <sub>14_2</sub>	W <sub>1_2</sub>	W <sub>2_2</sub>	b <sub>1_2</sub>	b <sub>1_2_2</sub>	b <sub>2_2</sub>



V <sub>11_2</sub>	V <sub>21_2</sub>	...	V <sub>141_2</sub>	V <sub>12_2</sub>	V <sub>22_2</sub>	...	V <sub>142_1</sub>	W <sub>1_1</sub>	W <sub>2_1</sub>	b <sub>1_1</sub>	b <sub>1_2_1</sub>	b <sub>2_1</sub>
V <sub>11_1</sub>	V <sub>21_1</sub>	...	V <sub>141_1</sub>	V <sub>12_1</sub>	V <sub>22_1</sub>	...	V <sub>14_2</sub>	W <sub>1_2</sub>	W <sub>2_2</sub>	b <sub>1_2</sub>	b <sub>1_2_2</sub>	b <sub>2_2</sub>

diberi nilai random bilangan uniform (0.1) di kromosom yang telah terpilih sebagai calon orang tua. Jika nilai random bilangan uniform (0.1) kurang dari peluang pindah silang ( $P_c=0.8$ ) sehingga kromosom tersebut terpilih menjadi orang tua dan terjadilah proses pindah silang yaitu kromosom satu melakukan pertukaran informasi (isi parameter) dengan kromosom 2 yang menghasilkan sebuah kromosom baru yang dihitung juga nilai fitnessnya.

Tahapan keenam yaitu mutasi dimana mutasi dilakukan pada kromosom yang terpilih dengan mengubah salah satu nilai parameter dengan suatu bilangan random. Kromosom dipilih dengan memberikan bilangan uniform (0.1) pada kromosom, jika bilangan random uniform (0.1) kurang dari nilai peluang mutasi ( $P_m = 0.1$ ) sehingga kromosom tersebut menjadi kromosom yang terpilih.

Tahapan ketujuh yaitu dengan mengurutkan nilai fitness dari semua kromosom yang terbentuk, dari 50 kromosom awal, kromosom hasil pindah silang, dan kromosom pada saat mutasi dari nilai fitness paling rendah sampai nilai fitness paling tinggi. Kromosom dengan nilai fitness tertinggi disimpan dan selanjutnya digunakan sebagai orang tua pada generasi selanjutnya. Tahapan berikutnya menggenerate 50 kromosom yang merupakan populasi baru dengan nilai fitness tertinggi dan dilakukan iterasi hingga iterasi ke 100.

Setelah mendapatkan parameter pada pelatihan *Backpropagation* yang paling optimal, kemudian dibentuk model untuk mengetahui kinerja klasifikasi pada data status penerimaan Bidikmisi. Tabel 4.26 merupakan 10-fold cross-validation hasil kinerja klasifikasi dari data status penerimaan beasiswa Bidikmisi. Berikut hasil pengoptimalan bobot dengan 4 neuron pada 1 hidden layer.

**Tabel 4.26** Hasil Kinerja Klasifikasi BPN-GA dengan 4 Neuron Pada 1 *Hidden Layer*

<i>Fold</i> <i>Cross ke-</i>	<i>Data Testing</i>			<i>Data Training</i>		
	AUC	G-mean	Akurasi	AUC	G-mean	Akurasi
1	0,50	0,00	96,62	0,51	0,00	96,63
2	0,51	0,13	96,05	0,50	0,08	96,09
3	0,52	0,00	96,60	0,50	0,00	96,65
4	0,53	0,11	96,66	0,52	0,04	96,59
5	0,48	0,00	96,62	0,53	0,00	96,64
6	0,50	0,00	96,64	0,51	0,00	96,64
7	0,52	0,00	96,43	0,52	0,00	96,65
8	0,50	0,18	93,34	0,51	0,21	93,28
9	0,50	0,00	96,62	0,51	0,00	96,63
10	0,52	0,50	39,11	0,51	0,50	39,31
Rata-rata	0,51	0,10	90,47	0,51	0,10	90,51

Hasil analisis pada Tabel 4.26 menunjukkan bahwa ketepatan klasifikasi pada data *training* adalah 90,51% dan data *testing* sebesar 90,47%, diketahui bahwa ketepatan klasifikasi dari model *Neural Network Backpropagation* yang terbentuk ditinjau dari akurasi sebesar 90,47%, ditinjau dari AUC sebesar 0,51 dan ditinjau dari G-mean sebesar 0,10. Dalam hal ini hasil dari data *training* dan data *testing* menghasilkan selisih yang hampir sama sehingga dapat dikatakan model yang dibentuk cukup baik.

#### 4.5 Tingkat Kinerja Klasifikasi *Neural Network Backpropagation* Tanpa dan dengan Optimasi Bobot dan Bias *Backpropagation*

Berikut akan diuraikan tentang perbandingan dari ketiga metode yang telah dilakukan, yaitu analisis klasifikasi dengan satu hidden layer, dan dengan dua hidden layer menggunakan *Neural Network Backpropagation* serta optimasi parameter bobot dan bias awal *Backpropagation*. Tabel 4.27 menyajikan Kinerja Hasil klasifikasi *Neural Network Backpropagation* Tanpa dan dengan Optimasi Bobot dan Bias *Backpropagation*.

**Tabel 4.27** Kinerja Hasil Klasifikasi Tanpa dan dengan Optimasi Bobot dan Bias Pada *Neural Network Backpropagation*

Struktur NN	Data <i>Testing</i>			Data <i>Training</i>		
	AUC	G-mean	Akurasi (%)	AUC	G-mean	Akurasi (%)
23-4-1 (Tanpa Optimasi)	0,51	0,01	78,00	0,51	0,01	78,02
<b>23-4-1 (Dengan Optimasi)</b>	<b>0,51</b>	<b>0,09</b>	<b>90,47</b>	<b>0,51</b>	<b>0,08</b>	<b>90,51</b>
23-10-4-1 (Tanpa Optimasi)	0,51	0,01	68,73	0,50	0,01	68,72

Berdasarkan Tabel 4.27 dapat diketahui bahwa bobot dan bias awal yang dioptimasi menggunakan Algoritma Genetika mampu meningkatkan hasil kinerja klasifikasi untuk AUC, *G-mean* dan Akurasi. Dapat dilihat hasil dari kinerja *Backpropagation* dengan struktur NN (23-4-1) sebelum dioptimasi atau parameter bobot dan bias yang dilakukan secara acak manual pada data *testing* menghasilkan nilai akurasi sebesar 78,00% , nilai *G-mean* sebesar 0,01 dan nilai AUC 0,51. Setelah dilakukan optimasi menggunakan Algoritma Genetika dihasilkan nilai akurasi sebesar 90,47%, nilai *G-mean* sebesar 0,09, nilai AUC sebesar 0,51 walaupun tidak meningkat secara signifikan tetapi dapat dikatakan bahwa optimasi bobot dan bias awal menggunakan algoritma genetika mampu meningkatkan kinerja klasifikasi dan untuk hasil data *testing* DLNN (23-10-4) menghasilkan nilai akurasi sebesar 68,73%, nilai *G-mean* sebesar 0,01, nilai AUC sebesar 0,51 sehingga dalam hal ini dapat dikatakan bahwa hasil klasifikasi yang terbaik yaitu klasifikasi dengan optimasi bobot dan bias pada *Neural Network Backpropagation*.

Berdasarkan analisis klasifikasi yang telah dibahas diketahui bahwa hasil nilai akurasi yang tinggi tetapi menghasilkan nilai *G-mean* yang rendah dikarenakan data yang tidak seimbang yaitu proporsi kelas status penerimaan Beasiswa Bidikmisi data positif (diterima) sebesar 0,96648 atau 96,648% sedangkan data negatif (tidak diterima) sebesar 0,0335 atau 3,35% dan jika dilihat dari Gambar 4.2 sebaran data status Beasiswa Bidikmisi Jawa Timur Tahun 2017 terjadi *overlapping* antar kelas.

*(halaman ini sengaja dikosongkan)*

## BAB 5

### KESIMPULAN DAN SARAN

#### 5,1 Kesimpulan

Dari hasil yang digunakan dalam penelitian ini dapat diambil beberapa kesimpulan sebagai berikut.

1. Model yang dibentuk pada *Neural Network Backpropagation* dengan 1 hidden layer menghasilkan prediksi status penerimaan Beasiswa Bidikmisi di Jawa Timur dengan akurasi sebesar 78,00%, nilai *G-mean* sebesar 0,01, nilai AUC 0,51 sedangkan untuk 2 hidden layer menghasilkan nilai akurasi sebesar 68,73%, nilai *G-mean* sebesar 0,01, nilai AUC sebesar 0,51.
2. Optimasi Parameter Bobot dan Bias *Backpropagation* mampu memperbaiki terhadap proses pelatihan *Neural Network* dengan meningkatkan kinerja klasifikasi *Backpropagation* yang dapat dianalisis dari hasil kinerja *Backpropagation* dengan arsitektur NN (23-4-1) sebelum dioptimasi atau parameter bobot dan bias yang dilakukan secara acak manual pada data *testing* menghasilkan nilai akurasi sebesar 78,00% , nilai *G-mean* sebesar 0,01 dan nilai AUC 0,51, Setelah dilakukan optimasi menggunakan Algoritma Genetika dihasilkan nilai akurasi sebesar 90,47%, nilai *G-mean* sebesar 0,10, nilai AUC sebesar 0,51 walaupun tidak meningkat secara signifikan tetapi dapat dikatakan bahwa optimasi bobot dan bias awal menggunakan algoritma genetika mampu meningkatkan kinerja klasifikasi.
3. Berdasarkan dari analisis yang telah diuraikan dapat disimpulkan hasil klasifikasi yang terbaik yaitu klasifikasi dengan optimasi bobot dan bias pada *Neural Network Backpropagation* karna dapat meningkatkan kinerja klasifikasi, yaitu hasil nilai akurasi sebesar 90,51%, nilai *G-mean* sebesar 0,10, dan nilai AUC sebesar 0,51.



## 5.1 Saran

1. Berdasarkan penelitian yang telah dilakukan, metode BPN-GA dapat digunakan untuk klasifikasi status penerimaan Beasiswa Bidikmisi di Jawa Timur karena pada penelitian ini metode BPN-GA menghasilkan nilai kinerja klasifikasi yang lebih baik dari metode lainnya.
2. Untuk penelitian selanjutnya dari analisis klasifikasi yang telah diuraikan menghasilkan rata-rata hasil sensitivity yang sama sebagian besar hampir semua kelas data positif (diterima Bidikmisi) yang diprediksi sebagai positif atau seluruh kelas 1 dapat diprediksi dengan benar, Sedangkan kelas 0 (Tidak diterima) tidak semuanya diprediksi dengan benar, Walau akurasi yang didapat 94,21% tetapi *classifier* kurang baik melakukan prediksi kelas negative. Hal ini dikarenakan jumlah data antar kelas positif dan negatif tidak seimbang seperti yang telah diuraikan sebelumnya proporsi kelas status penerimaan Beasiswa Bidikmisi data positif (diterima) sebesar 0,96648 atau 96,648% sedangkan data negatif (tidak diterima) sebesar 0,0335 atau 3,35% dan jika dilihat dari Gambar 4.2 sebaran data status Beasiswa Bidikmisi Jawa Timur Tahun 2017 dari gambar tersebut terjadi *overlapping* antar *class*. Sehingga untuk menyelesaikan masalah *imbalance* menggunakan metode *Undersampling* atau *Oversampling* dan dapat juga menggunakan Algoritma *Boosting* serta menyeleksi variabel input dengan menggunakan GA.

## DAFTAR PUSTAKA

- Ahmed, O., Nordin, M., Sulaiman, S., & Fatimah, W. (2009). Study of Genetic Algorithm to Fully-automate the Design and Training of Artificial Neural Network . *IJCSNS International Journal of Computer Science and Network Security* , 217-226.
- Arhami, M., & Desiani, A. (2006). *Konsep Kecerdasan Buatan*. Yogyakarta: Andi Offset.
- Departemen Pendidikan Nasional. (2010). *Program Beasiswa Bidik Misi:Beasiswa Pendidikan Bagi Calon Mahasiswa Berprestasi dari Keluarga Kurang Mampu*. Jakarta: Direktorat Jendral Pendidikan Tinggi.
- Desiani, A., & Arhami, M. (2006). *Konsep Kecerdasan Buatan*. Yogyakarta: Andi.
- Faisal, Reza.M (2008). *Seri Belajar Data Science Klasifikasi Dengan Bahasa Pemrograman R*. Banjarmasin.
- Gen, M., & Cheng. (1997). *Genetic Algorithms dan Engineering Design*. Canada: John Wiley dan Sons, Inc.
- Golberg, D. E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Canada: Addison Wesley Longman, Inc.
- Han, J., & Kamber, M. (2006). *Data mining : Concepts and Techniques*. California: Second Edition, Morgan Kaufmann.
- Haupt, S. E., & Haupt, R. L. (2004). *Partical Genetic Algorithms*. New Jersey: A John Wiley & Sons Inc.
- Karegowda, A. G., Manjunath, A. S., & Jayaram, M. A. (2011). Application of Genetic Algorithm Optimized Neural Network Connection Weights for Medical Diagnosis of PIMA Indian Diabetes. *IJSC International Journal on Soft Computing* .
- Kementrian Riset Teknologi & Pendidikan. (2015). *Pedoman Penyelenggaraan Bantuan Biaya Pendidikan Bidikmisi Tahun 2015*. Jakarta: Direktorat Jendral Pembelajaran dan Kemahasiswaan.
- Kohavi, R. (2011). A Study of Cross-Validation ang Bootstrap for Accuracy Estimation and Model Selection. The Fourth International Joint Conference on Artificial Intelligence. *American Association for Artificial Intelligence* , 202-209.

- Kusumadewi, S. (2004). *Membangun Jaringan Saraf Tiruan Menggunakan MATLAB & EXCEL LINK*. Yogyakarta: Graha Ilmu.
- Last, M. (2006). The Uncertainty Principle of Cross-Validation. *IEEE Conference Publications* , 275-280.
- Luthfi, E. M. (2009). *Algoritma Data Mining*. Yogyakarta: ANDI.
- Lye, C. T., Ng, L. N., Hassan, M. D., Goh, W. W., Law, C. Y., & Ismail, N. (2010). Predicting Pre-university Students' Mathematics Achievement. *Procedia Social and Behavioral Sciences* , 299-306.
- Oladokun, V. O., Adebajo, A. T., & Charles-Owaba, O. E. (2008). Predicting Students' Academic Performance using Artificial Neural Network: A Case Study of an Engineering Course. *The Pacific Journal of Science and Technology* .
- Prasetyo, E. (2006). *Data Mining Konsep dan Aplikasi Menggunakan MATLAB*. Yogyakarta: Andi.
- Santosa, B., & Willy, P. (2010). *Metode Metheuristik: Konsep dan Implementasi*. Surabaya: Guna Widya.
- Sayekti, I. (2013). Pengujian Model Jaringan Syaraf tiruan Untuk Kualifikasi Calon Mahasiswa baru Program Bidik Misi. *Jurnal teknik Elektro Terapan* , 55-60.
- Siang, J. J. (2005). *Jaringan Syaraf Tiruan dan Pemrogramannya Menggunakan Matlab*. Yogyakarta: Andi.
- Suyanto. (2005). *Algoritma Genetika dalam Matlab*. Yogyakarta: Andi Offset.
- Teshnizi, S. H., Taghi, S. M., & Ayatollahi. (2015). A Comparison of Logistic Regression Model and Artificial Neural Networks in Predicting of Student's Academic Failure. *Original Paper / Acta Inform Med* , 296-300.
- Trevino, V., & Falciani, F. (2006). An R Package for Multivariate Variable Selection Using Genetic Algorithms . *Bioinformatics* , 1154-1156.
- Turban, E., Aronson, J. E., & Liang, T. (2005). *Decision Support Systems and Intelligent Systems*. Yogyakarta: Andi Offset.
- Walczak, S., & Sincich, T. (Information Sciences). 1999. *Comparative analysis of regression and neural networks for university admission* , 1-20.

- Y.Du, & Y.Li. (2008). Sonar array azimuth control system based on genetic neural network. *Proceedings of the 7th World Congress on Intelligent Control and ,* pp. 6123-6127.
- Zukhri, Z. (2015). *Algoritma Genetika Metode Komputasi Evolusioner untuk Menyelesaikan Masalah Optimasi*. Yogyakarta: Andi.

*(halaman ini sengaja dikosongkan)*

**Lampiran A.** Data Penelitian Bidikmisi Provinsi Jawa Timur Tahun 2017

NO	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	Y
1	4	1	2	3	2	1	2	3	2	2	1	4	1
2	5	1	2	3	3	1	3	3	2	2	1	2	1
3	4	1	3	2	2	1	3	3	2	1	1	1	1
4	5	2	3	2	3	1	3	3	2	1	1	1	1
5	2	1	2	2	2	1	3	3	1	1	1	4	1
6	2	2	2	2	3	1	2	3	1	1	3	3	1
7	4	2	3	3	3	3	3	3	2	2	1	4	1
8	2	1	2	2	2	1	2	3	1	1	3	4	1
9	2	2	3	1	2	2	3	3	3	3	1	2	1
10	2	2	3	2	2	2	3	3	2	1	1	4	1
:	:	:	:	:	:	:	:	:	:	:	:	:	:
52411	2	2	3	2	3	1	3	3	2	1	1	3	1
52412	2	2	2	2	3	2	3	3	1	1	1	3	1
52413	5	4	3	2	3	2	3	3	2	1	1	1	1
52414	2	1	2	2	2	1	3	3	1	1	1	2	1
52415	6	1	5	4	1	1	3	3	3	2	1	2	1
52416	2	6	3	4	1	5	3	3	3	2	1	4	1
52417	1	1	3	3	1	1	2	3	1	1	1	2	1
52418	5	1	4	4	4	1	3	3	2	1	1	4	1
52419	4	1	3	1	3	1	3	3	3	3	1	2	0
52420	2	1	2	2	3	1	3	3	3	1	3	2	0

## Lampiran B. Lanjutan Syntax Membagi Data Menjadi 10-Fold Cross Validation

```
train_1= as.data.frame(rbind(train_1_yes,train_1_no))
test_1= as.data.frame(rbind(test_1_yes,test_1_no))

train_2_yes = Data_yes[-fold_yes[[2]],]
test_2_yes = Data_yes[fold_yes[[2]],]
train_2_no = Data_no[-fold_no[[2]],]
test_2_no = Data_no[fold_no[[2]],]
train_2= as.data.frame(rbind(train_2_yes,train_2_no))
test_2= as.data.frame(rbind(test_2_yes,test_2_no))

train_3_yes = Data_yes[-fold_yes[[3]],]
test_3_yes = Data_yes[fold_yes[[3]],]
train_3_no = Data_no[-fold_no[[3]],]
test_3_no = Data_no[fold_no[[3]],]
train_3= as.data.frame(rbind(train_3_yes,train_3_no))
test_3= as.data.frame(rbind(test_3_yes,test_3_no))

train_4_yes = Data_yes[-fold_yes[[4]],]
test_4_yes = Data_yes[fold_yes[[4]],]
train_4_no = Data_no[-fold_no[[4]],]
test_4_no = Data_no[fold_no[[4]],]
train_4= as.data.frame(rbind(train_4_yes,train_4_no))
test_4= as.data.frame(rbind(test_4_yes,test_4_no))

train_5_yes = Data_yes[-fold_yes[[5]],]
test_5_yes = Data_yes[fold_yes[[5]],]
train_5_no = Data_no[-fold_no[[5]],]
test_5_no = Data_no[fold_no[[5]],]
train_5= as.data.frame(rbind(train_5_yes,train_5_no))
test_5= as.data.frame(rbind(test_5_yes,test_5_no))

train_6_yes = Data_yes[-fold_yes[[6]],]
test_6_yes = Data_yes[fold_yes[[6]],]
train_6_no = Data_no[-fold_no[[6]],]
test_6_no = Data_no[fold_no[[6]],]
train_6= as.data.frame(rbind(train_6_yes,train_6_no))
test_6= as.data.frame(rbind(test_6_yes,test_6_no))

train_7_yes = Data_yes[-fold_yes[[7]],]
test_7_yes = Data_yes[fold_yes[[7]],]
```

## Lampiran B. Lanjutan Syntax Membagi Data Menjadi 10-Fold Cross Validation

```
train_7_no = Data_no[-fold_no[[7]],]
test_7_no = Data_no[fold_no[[7]],]
train_7= as.data.frame(rbind(train_7_yes,train_7_no))
test_7= as.data.frame(rbind(test_7_yes,test_7_no))

train_8_yes = Data_yes[-fold_yes[[8]],]
test_8_yes = Data_yes[fold_yes[[8]],]
train_8_no = Data_no[-fold_no[[8]],]
test_8_no = Data_no[fold_no[[8]],]
train_8= as.data.frame(rbind(train_8_yes,train_8_no))
test_8= as.data.frame(rbind(test_8_yes,test_8_no))

train_9_yes = Data_yes[-fold_yes[[9]],]
test_9_yes = Data_yes[fold_yes[[9]],]
train_9_no = Data_no[-fold_no[[9]],]
test_9_no = Data_no[fold_no[[9]],]
train_9= as.data.frame(rbind(train_9_yes,train_9_no))
test_9= as.data.frame(rbind(test_9_yes,test_9_no))

train_10_yes = Data_yes[-fold_yes[[10]],]
test_10_yes = Data_yes[fold_yes[[10]],]
train_10_no = Data_no[-fold_no[[10]],]
test_10_no = Data_no[fold_no[[10]],]
train_10= as.data.frame(rbind(train_10_yes,train_10_no))
test_10= as.data.frame(rbind(test_10_yes,test_10_no))

#Menyimpan Dataset Train dan Test per Fold
write.csv(train_1,"D:/Nita/data_train1.csv",row.names = FALSE)
write.csv(test_1,"D:/Nita/data_test1.csv",row.names = FALSE)
write.csv(train_2,"D:/Nita/data_train2.csv",row.names = FALSE)
write.csv(test_2,"D:/Nita/data_test2.csv",row.names = FALSE)
write.csv(train_3,"D:/Nita/data_train3.csv",row.names = FALSE)
write.csv(test_3,"D:/Nita/data_test3.csv",row.names = FALSE)
write.csv(train_4,"D:/Nita/data_train4.csv",row.names = FALSE)
write.csv(test_4,"D:/Nita/data_test4.csv",row.names = FALSE)
write.csv(train_5,"D:/Nita/data_train5.csv",row.names = FALSE)
write.csv(test_5,"D:/Nita/data_test5.csv",row.names = FALSE)
write.csv(train_6,"D:/Nita/data_train6.csv",row.names = FALSE)
write.csv(test_6,"D:/Nita/data_test6.csv",row.names = FALSE)
write.csv(train_7,"D:/Nita/data_train7.csv",row.names = FALSE)
```



## Lampiran B. Lanjutan Syntax Membagi Data Menjadi 10-Fold Cross Validation

```
write.csv(test_7,"D:/Nita/data_test7.csv",row.names = FALSE)
write.csv(train_8,"D:/Nita/data_train8.csv",row.names = FALSE)
write.csv(test_8,"D:/Nita/data_test8.csv",row.names = FALSE)
write.csv(train_9,"D:/Nita/data_train9.csv",row.names = FALSE)
write.csv(test_9,"D:/Nita/data_test9.csv",row.names = FALSE)
write.csv(train_10,"D:/Nita/data_train10.csv",row.names = FALSE)
write.csv(test_10,"D:/Nita/data_test10.csv",row.names = FALSE)
```

## Lampiran C. Syntax Neural Network 1 Hidden Layer 10-Fold Cross Validation

```
rm(list=ls())
library(readr)
library(neuralnet)
library(MLmetrics)
library(caret)
library(ROCR)

#Membuat Model Neural Network
Data_train1 =read_csv("D:/Nita/data_train1.csv")
Data_test1 =read_csv("D:/Nita/data_test1.csv")
Data_train2=read_csv("D:/Nita/data_train2.csv")
Data_test2=read_csv("D:/Nita/data_test2.csv")
Data_train3=read_csv("D:/Nita/data_train3.csv")
Data_test3=read_csv("D:/Nita/data_test3.csv")
Data_train4=read_csv("D:/Nita/data_train4.csv")
Data_test4=read_csv("D:/Nita/data_test4.csv")
Data_train5=read_csv("D:/Nita/data_train5.csv")
Data_test5=read_csv("D:/Nita/data_test5.csv")
Data_train6=read_csv("D:/Nita/data_train6.csv")
Data_test6=read_csv("D:/Nita/data_test6.csv")
Data_train7=read_csv("D:/Nita/data_train7.csv")
Data_test7=read_csv("D:/Nita/data_test7.csv")
Data_train8=read_csv("D:/Nita/data_train8.csv")
Data_test8=read_csv("D:/Nita/data_test8.csv")
Data_train9=read_csv("D:/Nita/data_train9.csv")
Data_test9=read_csv("D:/Nita/data_test9.csv")
Data_train10=read_csv("D:/Nita/data_train10.csv")
Data_test10=read_csv("D:/Nita/data_test10.csv")
```

## Lampiran C. Lanjutan Syntax Neural Network 1 Hidden Layer 10-Fold Cross Validation

```
set.seed(123)

Model_1_2<-
neuralnet(D.Y~X11+X12+X13+X14+X15+X21+X22+X23+X24+X25+V11+V12+V13+
V14+X71+X72+X81+X82+V19+V20+X111+X112+V23,data = Data_train1,
          linear.output=FALSE,hidden=6,lifesign = "full", algorithm = "backprop",
learningrate = 0.01, threshold = 1)
Model_2_2<-
neuralnet(D.Y~X11+X12+X13+X14+X15+X21+X22+X23+X24+X25+V11+V12+V13+
V14+X71+X72+X81+X82+V19+V20+X111+X112+V23,data = Data_train2,
          linear.output=FALSE,hidden=6, lifesign = "full", algorithm =
"backprop",learningrate = 0.01, threshold = 1)
Model_3_2<-
neuralnet(D.Y~X11+X12+X13+X14+X15+X21+X22+X23+X24+X25+V11+V12+V13+
V14+X71+X72+X81+X82+V19+V20+X111+X112++V23,data = Data_train3,
          linear.output=FALSE,hidden=6,lifesign = "full", algorithm = "backprop",
learningrate = 0.01, threshold = 1)
Model_4_2<-neuralnet(D.Y~X11+X12+X13+X14+X15+X21+
X22+X23+X24+X25+V11+V12+V13+V14+X71+X72+X81+X82+V19+V20+
X111+X112+V23,data = Data_train4,
          linear.output=FALSE,hidden=6,lifesign = "full", algorithm = "backprop",
learningrate = 0.01, threshold = 1)
Model_5_2<-neuralnet(D.Y~X11+X12+X13+X14+X15+X21+
X22+X23+X24+X25+V11+V12+V13+V14+X71+X72+X81+X82+V19+V20+
X111+X112+V23,data = Data_train5,
          linear.output=FALSE,hidden=6,lifesign = "full", algorithm = "backprop",
learningrate = 0.01, threshold = 1)
Model_6_2<-neuralnet(D.Y~X11+X12+X13+X14+X15+X21+
X22+X23+X24+X25+V11+V12+V13+V14+X71+X72+X81+X82+V19+V20+
X111+X112+V23,data = Data_train6,
          linear.output=FALSE,hidden=6,lifesign = "full", algorithm = "backprop",
learningrate = 0.01, threshold = 1)
Model_7_2<-neuralnet(D.Y~X11+X12+X13+X14+X15+X21+
X22+X23+X24+X25+V11+V12+V13+V14+X71+X72+X81+X82+V19+V20+
X111+X112+V23,data = Data_train7,
          linear.output=FALSE,hidden=6,lifesign = "full", algorithm = "backprop",
learningrate = 0.01, threshold = 1)
Model_8_2<-neuralnet(D.Y~X11+X12+X13+X14+X15+X21+
X22+X23+X24+X25+V11+V12+V13+V14+X71+X72+X81+X82+V19+V20+
X111+X112+V23,data = Data_train8,
          linear.output=FALSE,hidden=6,lifesign = "full", algorithm = "backprop",
learningrate = 0.01, threshold = 1)
Model_9_2<-neuralnet(D.Y~X11+X12+X13+X14+X15+X21+
X22+X23+X24+X25+V11+V12+V13+V14+X71+X72+X81+X82+V19+V20+
X111+X112+V23,data = Data_train9,
          linear.output=FALSE,hidden=6,lifesign = "full", algorithm = "backprop",
learningrate = 0.01, threshold = 1)
Model_10_2<-
neuralnet(D.Y~X11+X12+X13+X14+X15+X21+X22+X23+X24+X25+V11+V12+V13+
V14+X71+X72+X81+X82+V19+V20+X111+X112+V23,data = Data_train10,
```

**Lampiran C.** Lanjutan Syntax Neural Network 1 Hidden Layer 10-Fold Cross  
*Validation*

```
Predict_1_2=compute(Model_1_2,Data_test1[,1:23])$net.result
Predict_2_2=compute(Model_2_2,Data_test2[,1:23])$net.result
Predict_3_2=compute(Model_3_2,Data_test3[,1:23])$net.result
Predict_4_2=compute(Model_4_2,Data_test4[,1:23])$net.result
Predict_5_2=compute(Model_5_2,Data_test5[,1:23])$net.result
Predict_6_2=compute(Model_6_2,Data_test6[,1:23])$net.result
Predict_7_2=compute(Model_7_2,Data_test7[,1:23])$net.result
Predict_8_2=compute(Model_8_2,Data_test8[,1:23])$net.result
Predict_9_2=compute(Model_9_2,Data_test9[,1:23])$net.result
Predict_10_2=compute(Model_10_2,Data_test10[,1:23])$net.result
library(InformationValue)
optCutOff1 <- optimalCutoff(Data_test1$D.Y, Predict_1_2)[1]
optCutOff2 <- optimalCutoff(Data_test2$D.Y, Predict_2_2)[1]
optCutOff3 <- optimalCutoff(Data_test3$D.Y, Predict_3_2)[1]
optCutOff4 <- optimalCutoff(Data_test4$D.Y, Predict_4_2)[1]
optCutOff5 <- optimalCutoff(Data_test5$D.Y, Predict_5_2)[1]
optCutOff6 <- optimalCutoff(Data_test6$D.Y, Predict_6_2)[1]
optCutOff7 <- optimalCutoff(Data_test7$D.Y, Predict_7_2)[1]
optCutOff8 <- optimalCutoff(Data_test8$D.Y, Predict_8_2)[1]
optCutOff9 <- optimalCutoff(Data_test9$D.Y, Predict_9_2)[1]
optCutOff10 <- optimalCutoff(Data_test10$D.Y,Predict_10_2)[1]
optCutOff1
optCutOff2
optCutOff3
optCutOff4
optCutOff5
optCutOff6
optCutOff7
optCutOff8
optCutOff9
optCutOff10
misClassError_test1 <- misClassError(Data_test1$D.Y, Predict_1_2, threshold =
optCutOff1)
misClassError_test2 <- misClassError(Data_test2$D.Y, Predict_2_2, threshold =
optCutOff2)
misClassError_test3 <- misClassError(Data_test3$D.Y, Predict_3_2, threshold =
optCutOff3)
misClassError_test4 <- misClassError(Data_test4$D.Y, Predict_4_2, threshold =
optCutOff4)
misClassError_test5 <-misClassError(Data_test5$D.Y, Predict_5_2, threshold =
optCutOff5)
misClassError_test6 <-misClassError(Data_test6$D.Y, Predict_6_2, threshold =
optCutOff6)
misClassError_test7 <-misClassError(Data_test7$D.Y, Predict_7_2, threshold =
optCutOff7)
misClassError_test8 <-misClassError(Data_test8$D.Y, Predict_8_2, threshold =
optCutOff8)
misClassError_test9 <-misClassError(Data_test9$D.Y, Predict_9_2, threshold =
optCutOff9)
```

## Lampiran C. Lanjutan Syntax Neural Network 1 Hidden Layer 10-Fold Cross Validation

```
misClassError_test10 <- misClassError(Data_test10$D.Y, Predict_10_2, threshold =
optCutOff10)
Accuracy_test1 = 1-misClassError_test1
Accuracy_test2 = 1-misClassError_test2
Accuracy_test3 = 1-misClassError_test3
Accuracy_test4 = 1-misClassError_test4
Accuracy_test5 = 1-misClassError_test5
Accuracy_test6 = 1-misClassError_test6
Accuracy_test7 = 1-misClassError_test7
Accuracy_test8 = 1-misClassError_test8
Accuracy_test9 = 1-misClassError_test9
Accuracy_test10 = 1-misClassError_test10
Accuracy_rata <-
(Accuracy_test1+Accuracy_test2+Accuracy_test3+Accuracy_test4+Accuracy_test5+Acc
uracy_test6+Accuracy_test7+Accuracy_test8+Accuracy_test9+Accuracy_test10)/10
#ROC 1
ROC1 <- plotROC(Data_test1$D.Y, Predict_1_2)
sensitivity1 <- sensitivity(Data_test1$D.Y, Predict_1_2, threshold = optCutOff1)
specificity1 <- specificity(Data_test1$D.Y, Predict_1_2, threshold = optCutOff1)
Gmean_Test1=sqrt(sensitivity1*specificity1)
confusionMatrix1 <- confusionMatrix(Data_test1$D.Y, Predict_1_2, threshold =
optCutOff1)
#ROC 2
ROC2 <- plotROC(Data_test2$D.Y, Predict_2_2)
sensitivity2 <- sensitivity(Data_test2$D.Y, Predict_2_2, threshold = optCutOff2)
specificity2 <- specificity(Data_test2$D.Y, Predict_2_2, threshold = optCutOff2)
Gmean_Test2=sqrt(sensitivity2*specificity2)
confusionMatrix2 <- confusionMatrix(Data_test2$D.Y, Predict_2_2, threshold =
optCutOff2)
#ROC 3
ROC3 <- plotROC(Data_test3$D.Y, Predict_3_2)
sensitivity3 <- sensitivity(Data_test3$D.Y, Predict_3_2, threshold = optCutOff3)
specificity3 <- specificity(Data_test3$D.Y, Predict_3_2, threshold = optCutOff3)
Gmean_Test3=sqrt(sensitivity3*specificity3)
confusionMatrix3 <- confusionMatrix(Data_test3$D.Y, Predict_3_2, threshold =
optCutOff3)
#ROC 4
ROC4 <- plotROC(Data_test4$D.Y, Predict_4_2)
sensitivity4 <- sensitivity(Data_test4$D.Y, Predict_4_2, threshold = optCutOff4)
specificity4 <- specificity(Data_test4$D.Y, Predict_4_2, threshold = optCutOff4)
Gmean_Test4=sqrt(sensitivity4*specificity4)
confusionMatrix4 <- confusionMatrix(Data_test4$D.Y, Predict_4_2, threshold =
optCutOff4)
#ROC 5
ROC5 <- plotROC(Data_test5$D.Y, Predict_5_2)
sensitivity5 <- sensitivity(Data_test5$D.Y, Predict_5_2, threshold = optCutOff5)
specificity5 <- specificity(Data_test5$D.Y, Predict_5_2, threshold = optCutOff5)
Gmean_Test5=sqrt(sensitivity5*specificity5)
```

## Lampiran C. Lanjutan Syntax Neural Network 1 Hidden Layer 10-Fold Cross Validation

```
confusionMatrix5 <- confusionMatrix(Data_test5$D.Y, Predict_5_2, threshold =
optCutOff5)
#ROC 6
ROC6 <- plotROC(Data_test6$D.Y, Predict_6_2)
sensitivity6 <- sensitivity(Data_test6$D.Y, Predict_6_2, threshold = optCutOff6)
specificity6 <- specificity(Data_test6$D.Y, Predict_6_2, threshold = optCutOff6)
Gmean_Test6=sqrt(sensitivity6*specificity6)
confusionMatrix6 <- confusionMatrix(Data_test6$D.Y, Predict_6_2, threshold =
optCutOff6)
#ROC 7
ROC7 <- plotROC(Data_test7$D.Y, Predict_7_2)
sensitivity7 <- sensitivity(Data_test7$D.Y, Predict_7_2, threshold = optCutOff7)
specificity7 <- specificity(Data_test7$D.Y, Predict_7_2, threshold = optCutOff7)
Gmean_Test7=sqrt(sensitivity7*specificity7)
confusionMatrix7 <- confusionMatrix(Data_test7$D.Y, Predict_7_2, threshold =
optCutOff7)
#ROC 8
ROC8 <- plotROC(Data_test8$D.Y, Predict_8_2)
sensitivity8 <- sensitivity(Data_test8$D.Y, Predict_8_2, threshold = optCutOff8)
specificity8 <- specificity(Data_test8$D.Y, Predict_8_2, threshold = optCutOff8)
Gmean_Test8=sqrt(sensitivity8*specificity8)
confusionMatrix8 <- confusionMatrix(Data_test8$D.Y, Predict_8_2, threshold =
optCutOff8)
#ROC 9
ROC9 <- plotROC(Data_test9$D.Y, Predict_9_2)
sensitivity9 <- sensitivity(Data_test9$D.Y, Predict_9_2, threshold = optCutOff9)
specificity9 <- specificity(Data_test9$D.Y, Predict_9_2, threshold = optCutOff9)
Gmean_Test9=sqrt(sensitivity9*specificity9)
confusionMatrix9 <- confusionMatrix(Data_test9$D.Y, Predict_9_2, threshold =
optCutOff9)
#ROC 10
ROC10 <- plotROC(Data_test10$D.Y, Predict_10_2)
sensitivity10 <- sensitivity(Data_test10$D.Y, Predict_10_2, threshold = optCutOff10)
specificity10 <- specificity(Data_test10$D.Y, Predict_10_2, threshold = optCutOff10)
Gmean_Test10=sqrt(sensitivity10*specificity10)
confusionMatrix10 <- confusionMatrix(Data_test10$D.Y, Predict_10_2,threshold =
optCutOff10)
#NilaiAUC
AUC1_1=AUC(Predict_1_2, Data_test1$D.Y)
AUC2_1=AUC(Predict_2_2, Data_test2$D.Y)
AUC3_1=AUC(Predict_3_2, Data_test3$D.Y)
AUC4_1=AUC(Predict_4_2, Data_test4$D.Y)
AUC5_1=AUC(Predict_5_2, Data_test5$D.Y)
AUC6_1=AUC(Predict_6_2, Data_test6$D.Y)
AUC7_1=AUC(Predict_7_2, Data_test7$D.Y)
AUC8_1=AUC(Predict_8_2, Data_test8$D.Y)
AUC9_1=AUC(Predict_9_2, Data_test9$D.Y)
AUC10_1=AUC(Predict_10_2, Data_test10$D.Y)

Gmean_Rata2 =
(Gmean_Test1+Gmean_Test2+Gmean_Test3+Gmean_Test4+Gmean_Test5+
```

### Lampiran C. Lanjutan Syntax Neural Network 1 Hidden Layer 10-Fold Cross Validation

```
Gmean_Rata2 =
(Gmean_Test1+Gmean_Test2+Gmean_Test3+Gmean_Test4+Gmean_Test5+
Gmean_Test6+Gmean_Test7+Gmean_Test8+Gmean_Test9+Gmean_Test10)/10
AUC_Rata2=(AUC1_1+AUC2_1+AUC3_1+AUC4_1+AUC5_1+AUC6_1+AUC7_1+A
UC8_1+AUC9_1+AUC10_1)/10

confusionMatrix1
confusionMatrix2
confusionMatrix3
confusionMatrix4
confusionMatrix5
confusionMatrix6
confusionMatrix7
confusionMatrix8
confusionMatrix9
confusionMatrix10

Gmean_Rata2
AUC_Rata2
Accuracy_rata
```

### Lampiran D. Syntax Optimasi Bobot Backpropagation dengan 1Hidden Layer dengan 4 Neuron

```
rm(list=ls())
library(readr)
library(GA)
library(neuralnet)
library(caret)
library(InformationValue)
trngdata = read_csv("D:/Nita/data_train10.csv")
tmgdata = read_csv("D:/Nita/data_test10.csv")
fitnessFunc <- function(x){
  # Retrieve the NN parameters
  model <-
neuralnet(D.Y~X11+X12+X13+X14+X15+X21+X22+X23+X24+X25+V11+V12+V13+
V14+X71+X72+X81+X82+V19+V20+X111+X112+V23,data = trngdata,
          linear.output=FALSE, hidden = 4,lifesign = "full", algorithm = "backprop",
learningrate = 0.01, threshold = 1)
b11_11<- x[1]
W11_21<- x[2]
W11_31<- x[3]
```

**Lampiran D.** Lanjutan Syntax Optimasi Bobot Backpropagation dengan  
*1Hidden Layer* dengan 4 Neuron

```
W11_41<- x[4]
W11_51<- x[5]
W11_61<- x[6]
W11_71<- x[7]
W11_81<- x[8]
W11_91<- x[9]
:
W11_221<- x[22]
W11_231<- x[23]
W11_241<- x[24]

b11_12 <- x[25]
W11_22 <- x[26]
W11_32 <- x[27]
W11_42<- x[28]
W11_52<- x[29]
W11_62<- x[30]
:
W11_212<- x[45]
W11_222<- x[46]
W11_232<- x[47]
W11_242<- x[48]

b11_13<- x[49]
W11_23 <- x[50]
:
W11_233<- x[71]
W11_243<- x[72]

b11_14<-x[73]
W11_24 <- x[74]
W11_34 <- x[75]
:
W11_224<- x[94]
W11_234<- x[95]
W11_244<- x[96]

b12_11 <- x[97]
W12_21<- x[98]
W12_31<- x[99]
W12_41<- x[100]
W12_51<- x[101]
```

## Lampiran D. Lanjutan Syntax Optimasi Bobot Backpropagation dengan

### *1Hidden Layer* dengan 4 Neuron

```
w_in=matrix(c(b11_11 , W11_21 , W11_31 , W11_41, W11_51, W11_61, W11_71 , W11_81 , W11_91 ,
W11_101, W11_111 , W11_121 , W11_131 , W11_141 , W11_151 , W11_161 ,
W11_171 , W11_181 , W11_191 , W11_201 , W11_211 , W11_221 , W11_231 ,
W11_241 ,
b11_12 , W11_22 , W11_32 , W11_42, W11_52, W11_62, W11_72 , W11_82 , W11_92 ,
W11_102, W11_112 , W11_122 , W11_132 , W11_142 , W11_152 , W11_162 ,
W11_172 , W11_182 , W11_192 , W11_202 , W11_212 , W11_222 , W11_232 ,
W11_242 ,
b11_13 , W11_23 , W11_33 , W11_43, W11_53, W11_63, W11_73 , W11_83 , W11_93 ,
W11_103, W11_113 , W11_123 , W11_133 , W11_143 , W11_153 , W11_163 ,
W11_173 , W11_183 , W11_193 , W11_203 , W11_213 , W11_223 , W11_233 ,
W11_243 ,
b11_14 , W11_24 , W11_34 , W11_44, W11_54, W11_64, W11_74 , W11_84 , W11_94 ,
W11_104, W11_114 , W11_124 , W11_134 , W11_144 , W11_154 , W11_164 ,
W11_174 , W11_184 , W11_194 , W11_204 , W11_214 , W11_224 , W11_234 ,
W11_244),24,4)
w_out=matrix(c(b12_11 , W12_21 , W12_31 , W12_41 , W12_51),5,1)

model$startweights[[1]][[1]]=w_in

model$startweights[[1]][[2]]=w_out

Pred1 =compute(model,tmgdata[,1:23])$net.result
AUC=AUC(Pred1,tmgdata$D.Y)
return(AUC)
}
theta_min <- c(b11_11 = -30, W11_21 = -30, W11_31 = -30, W11_41= -30, W11_51= 30, W11_61=30,
W11_71 = -30, W11_81 = -30 , W11_91 = -30, ... ,b12_11 = -30, W12_21 = -30, W12_31 = -30, W12_41
= -30, W12_51 = -30)

theta_max <- c(b11_11 = 30, W11_21 = 30, W11_31 = 30, W11_41= 30, W11_51= 30, W11_61=3 0,
W11_71 = 30, W11_81 = 30 , W11_91 = 30,..., b12_11 = 30, W12_21 = 30, W12_31 = 30, W12_41 = 30,
W12_51 = 30)
```



**Lampiran D.** Lanjutan Syntax Optimasi Bobot Backpropagation dengan  
*1Hidden Layer* dengan 4 Neuron

```
result = ga(type="real-valued",
            fitness=fitnessFunc,
            names=names(theta_min),
            min=theta_min,
            max=theta_max,
            popSize=50,
            maxiter=100,
            pcrossover=0.8,
            pmutation=0.1,
            monitor= plot)
hasil <- summary(result)[11]
fitnessvalue=summary(result)[10]
summary(result)
hasil$solution
plot(result)

set.seed(123)
#training BPN-GA
Model_BPNGA<-
neuralnet(D.Y~X11+X12+X13+X14+X15+X21+X22+X23+X24+X25+V11+V12+V13+V14+X71+X72+
X81+X82+V19+V20+X111+X112+V23,data = tmgdata,
          linear.output=FALSE,hidden=4,lifesign = "full", algorithm = "backprop", learningrate = 0.01,
          startweights = hasil$solution, threshold = 1)

Predict_BPNGA=compute(Model_BPNGA,tmgdata[,1:23])$net.result
library(InformationValue)
optCutOff1 <- optimalCutoff(tmgdata$D.Y, Predict_BPNGA)[1]
optCutOff1
misClassError_test1 <- misClassError(tmgdata$D.Y, Predict_BPNGA, threshold = optCutOff1)
Accuracy_test1 = 1-misClassError_test1
#ROC Data Testing
ROC1 <- plotROC(tmgdata$D.Y, Predict_BPNGA)
sensitivity1 <- sensitivity(tmgdata$D.Y, Predict_BPNGA, threshold = optCutOff1)
specificity1 <- specificity(tmgdata$D.Y, Predict_BPNGA, threshold = optCutOff1)
Gmean_Test1=sqrt(sensitivity1*specificity1)
confusionMatrix1 <- confusionMatrix(tmgdata$D.Y, Predict_BPNGA, threshold = optCutOff1)
#Nilai AUC Data Testing
AUC1_2=AUC(Predict_BPNGA, tmgdata$D.Y)
#Hasil Performace Ketepatan Klasifikasi BPNGA Data Testing
confusionMatrix1
sensitivity1
specificity1
Gmean_Test1
AUC1_2
Accuracy_test1
```

**Lampiran D.** Lanjutan Syntax Optimasi Bobot Backpropagation dengan  
*1Hidden Layer* dengan 4 Neuron

```
#train
Predict_BPNGA1=compute(Model_BPNGA,trngdata[,1:23])$net.result

library(InformationValue)
optCutOff1 <- optimalCutoff(trngdata$D.Y, Predict_BPNGA1)[1]
optCutOff1
misClassError_train1 <- misClassError(trngdata$D.Y, Predict_BPNGA1, threshold = optCutOff1)
Accuracy_train1 = 1-misClassError_train1

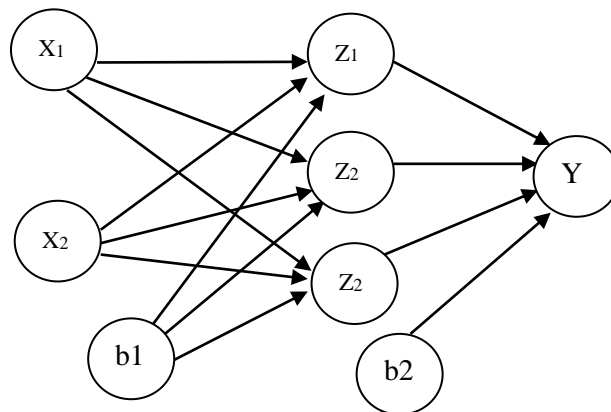
#ROC Data Training
ROC1 <- plotROC(trngdata$D.Y, Predict_BPNGA1)
sensitivity1 <- sensitivity(trngdata$D.Y, Predict_BPNGA1, threshold = optCutOff1)
specificity1 <- specificity(trngdata$D.Y, Predict_BPNGA1, threshold = optCutOff1)
Gmean_train1=sqrt(sensitivity1*specificity1)
confusionMatrix1 <- confusionMatrix(trngdata$D.Y, Predict_BPNGA1, threshold = optCutOff1)
#Nilai AUC Data Traininh
AUC1_2=AUC(Predict_BPNGA1,trngdata$D.Y)
#Hasil Performace Ketepatan Klasifikasi BPNGA Data Training
confusionMatrix1
sensitivity1
specificity1
Gmean_train1
Accuracy_train1
AUC1_2
```

**Lampiran E.** Jaringan Syaraf Tiruan *Backpropagation* (Contoh Manual Algoritma *Backpropagation*)

Misalnya diberikan data seperti pada table berikut, terdiri dari dua variabel input dengan 1 katagori atau kelas.

$X_1$	$X_2$	Y
1	1	0

Akan dibangun jaringan *backpropagation* dengan 3 *node* pada *hidden layer*. Maka arsitektur jaringan adalah sebagai berikut.



Langkah 0 : Inisialisasi bobot dengan bilangan yang sangat kecil

Bobot input ke *hidden layer*

	$Z_1$	$Z_2$	$Z_3$
$X_1$	0,2	0,3	-0,1
$X_2$	0,3	0,1	-0,1
$b_1$	-0,3	0,3	0,3

Bobot *hidden layer* ke output layer

	Y
$z_1$	0,5
$z_2$	-0,3
$z_3$	-0,4
$b_2$	-0,1

Langkah 1 : Ketika kondisi berhenti tidak terpenuhi, dilakukan langkah 2.

Langkah 2 : Untuk tiap tiap pasangan elemen yang akan dilakukan pembelajaran, kerjakan langkah-langkah berikutnya.

**Feedforward:**

- Tiap-tiap unit *input* ( $X_i, i = 1, 2, \dots, n$ ) menerima sinyal  $X_i$  dan meneruskan sinyal tersebut ke semua unit lapisan tersembunyi.
- Tiap-tiap unit pada suatu lapisan tersembunyi ( $Z_j, j = 1, 2, \dots, p$ ) menjumlahkan sinyal sinyal *input* terbobot,

**Lampiran E.** Lanjutan Jaringan Syaraf Tiruan *Backpropagation* (Contoh Manual Algoritma *Backpropagation*)

$$z_{in_j} = b1_j + \sum_{i=1}^n x_i v_{ij}$$

$$z_{in_1} = -0,3 + (1 \times 0,2) + (1 \times 0,3) = 0,2$$

$$z_{in_2} = 0,3 + (1 \times 0,3) + (1 \times 0,1) = 0,7$$

$$z_{in_3} = 0,3 + (1 \times -0,1) + (1 \times -0,1) = 0,1$$

menerapkan fungsi aktivasi untuk menghitung sinyal *output*,

$$z_j = f(z_{in_j}) = \frac{1}{1 + e^{-z_{in_j}}}$$

$$z_1 = f(z_{in_1}) = \frac{1}{1 + e^{-z_{in_1}}} = \frac{1}{1 + e^{-0,2}} = 0,55$$

$$z_2 = f(z_{in_2}) = \frac{1}{1 + e^{-z_{in_2}}} = \frac{1}{1 + e^{-0,7}} = 0,67$$

$$z_3 = f(z_{in_3}) = \frac{1}{1 + e^{-z_{in_3}}} = \frac{1}{1 + e^{-0,1}} = 0,53$$

dan mengirimkan sinyal *output* tersebut sebagai *input* untuk semua unit lapisan *output*.

- c. Tiap-tiap *output*  $Y_k$  ( $k = 1, 2, \dots, m$ ) menjumlahkan sinyal-sinyal *input* yang terboboti,

$$y_{in_k} = b2_k + \sum_{i=1}^p z_i w_{jk}$$

$$y_{in_1} = -0,1 + (0,55 \times 0,5) + (0,67 \times -0,3) + (0,53 \times -0,4) = -0,24$$

menerapkan fungsi aktivasi untuk menghitung sinyal *output*,

$$y_k = f(y_{in_k})$$

$$y_1 = f(y_{in_1}) = \frac{1}{1 + e^{-y_{in_1}}} = \frac{1}{1 + e^{-0,24}} = 0,44$$

**Lampiran E.** Lanjutan Jaringan Syaraf Tiruan *Backpropagation* (Contoh Manual Algoritma *Backpropagation*)

***Backpropagation:***

- d. Tiap-tiap unit *output*  $Y_k (k = 1, 2, \dots, m)$  menerima target pola yang berhubungan dengan pola *input* pelatihan, kemudian dihitung informasi *error*nya,

$$\begin{aligned}\delta 2_k &= (t_k - y_k)f'(y_{in_k}) \\ \delta 2_1 &= (t_1 - y_1)f'(y_{in_1}) \\ \delta 2_1 &= (t_1 - y_1)y_1(1 - y_1) = (0 - 0,44)0,44(1 - 0,44) = -0,108\end{aligned}$$

menghitung besar koreksi bobot yang digunakan untuk mengupdate informasi *error*nya,

$$\begin{aligned}\Delta w_{jk} &= \alpha \delta_k z_j \\ \Delta w_{11} &= \alpha \delta_1(z_1) = 0,2 \times -0,108 \times 0,55 = -0,01 \\ \Delta w_{12} &= \alpha \delta_1(z_2) = 0,2 \times -0,108 \times 0,67 = -0,01 \\ \Delta w_{13} &= \alpha \delta_1(z_3) = 0,2 \times -0,108 \times 0,53 = -0,01\end{aligned}$$

kemudian menghitung koreksi bias yang digunakan untuk mengupdate bias  $b2_k$ ,

$$\begin{aligned}\Delta b2_k &= \alpha \delta_k \\ \Delta b2_1 &= \alpha \delta_1(1) = 0,2 \times -0,108 = -0,02\end{aligned}$$

- e. Tiap-tiap unit tersembunyi  $Z_j (j = 1, 2, \dots, p)$  menjumlahkan delta *input*nya dari unit-unit lapisan *output*,

$$\begin{aligned}\delta_{in_j} &= \sum_{k=1}^m \delta 2_k w_{jk} \\ \delta_{in_1} &= -0,108 \times 0,5 = -0,05 \\ \delta_{in_2} &= -0,108 \times -0,3 = 0,03 \\ \delta_{in_3} &= -0,108 \times -0,4 = 0,04\end{aligned}$$

kalikan nilai ini dengan turunan dari fungsi aktivasinya untuk menghitung *error*,

$$\delta 1_j = \delta_{in_j} f'(z_{in_j})$$

**Lampiran E.** Lanjutan Jaringan Syaraf Tiruan *Backpropagation* (Contoh Manual Algoritma *Backpropagation*)

$$\delta 1_1 = \delta_{in_1} z_1 (1 - z_1) = -0,05 \times 0,55 \times (1 - 0,55) = -0,01$$

$$\delta 1_2 = \delta_{in_2} z_2 (1 - z_2) = 0,03 \times 0,67 \times (1 - 0,67) = 0,01$$

$$\delta 1_3 = \delta_{in_3} z_3 (1 - z_3) = 0,04 \times 0,53 \times (1 - 0,53) = 0,01$$

menghitung besar koreksi bobot yang digunakan untuk mengupdate bobot  $v_{ij}$ ,

$$\Delta v_{ij} = \alpha \delta 1_j x_j$$

$$\Delta v_{11} = 0,2 \times -0,01 \times 1 = -0,002$$

$$\Delta v_{21} = 0,2 \times -0,01 \times 1 = -0,002$$

$$\Delta v_{12} = 0,2 \times 0,01 \times 1 = 0,002$$

$$\Delta v_{22} = 0,2 \times 0,01 \times 1 = 0,002$$

$$\Delta v_{13} = 0,2 \times 0,01 \times 1 = 0,002$$

$$\Delta v_{23} = 0,2 \times 0,01 \times 1 = 0,002$$

menghitung koreksi bias yang digunakan untuk mengupdate bias  $b1_j$ ,

$$\Delta b1_j = \alpha \delta 1_j$$

$$\Delta b1_1 = 0,2 \times -0,01 = -0,002$$

$$\Delta b1_2 = 0,2 \times 0,01 = 0,002$$

$$\Delta b1_3 = 0,2 \times 0,01 = 0,002$$

- f. Tiap-tiap unit *output* ( $Y_k, k = 1, 2, 3, \dots, m$ ) mengupdate bias dan bobotnya ( $j = 1, 2, \dots, p$ ),

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta w_{jk}$$

$$w_{11}(\text{baru}) = w_{11}(\text{lama}) + \Delta w_{11} = 0,5 + (-0,012) = 0,488$$

$$w_{21}(\text{baru}) = w_{21}(\text{lama}) + \Delta w_{21} = -0,3 + (-0,015) = -0,315$$

$$w_{31}(\text{baru}) = w_{31}(\text{lama}) + \Delta w_{31} = -0,4 + (-0,011) = -0,411$$

$$b2_k(\text{baru}) = b2_k(\text{lama}) + \Delta b2_k$$

$$b2_1(\text{baru}) = b2_1(\text{lama}) + \Delta b2_1 = -0,1 + (-0,02) = -0,122$$

Tiap-tiap unit tersembunyi ( $Z_j, j = 1, 2, 3, \dots, p$ ) mengupdate bias dan bobotnya ( $i = 0, 1, 2, \dots, n$ ),

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij}$$

**Lampiran E.** Lanjutan Jaringan Syaraf Tiruan *Backpropagation* (Contoh Manual Algoritma *Backpropagation*)

$$v_{11}(\text{baru}) = v_{11}(\text{lama}) + \Delta v_{11} = 0,2 + (-0,002) = 0,198$$

$$v_{12}(\text{baru}) = v_{12}(\text{lama}) + \Delta v_{12} = 0,3 + (0,002) = 0,302$$

$$v_{13}(\text{baru}) = v_{13}(\text{lama}) + \Delta v_{13} = -0,1 + (0,002) = -0,098$$

$$v_{21}(\text{baru}) = v_{21}(\text{lama}) + \Delta v_{21} = 0,3 + (-0,002) = 0,298$$

$$v_{22}(\text{baru}) = v_{22}(\text{lama}) + \Delta v_{22} = 0,1 + (0,002) = 0,102$$

$$v_{23}(\text{baru}) = v_{23}(\text{lama}) + \Delta v_{23} = -0,1 + (0,002) = -0,098$$

$$b1_j(\text{baru}) = b1_j(\text{lama}) + \Delta b1_j$$

$$b1_1(\text{baru}) = b1_1(\text{lama}) + \Delta b1 = -0,3 + (-0,002) = -0,302$$

$$b1_2(\text{baru}) = b1_2(\text{lama}) + \Delta b1 = 0,3 + (-0,002) = 0,298$$

$$b1_3(\text{baru}) = b1_3(\text{lama}) + \Delta b1 = 0,3 + (-0,002) = 0,298$$

*(halaman ini sengaja dikosongkan)*



## BIOGRAFI PENULIS



[cahyani.nc@gmail.com](mailto:cahyani.nc@gmail.com)

Penulis dilahirkan di Seririt, Bali merupakan anak ke-2 dari 4 bersaudara, pasangan Endrian Suryana dan Mahdiyanah. Penulis telah menempuh pendidikan formal sebagai berikut: SD MII Maya Seririt (Tahun 1995-2001); SMP Negeri 1 Seririt (Tahun 2001-2004); SMA Negeri 8 Bogor (2004); SMA Negeri 6 Bogor (Tahun 2005-2007); Jurusan Statistika, Fakultas FMKSD (Fakultas Matematika, Komputasi, Dan Sains Data)

Institut Teknologi Sepuluh Nopember Surabaya (Tahun 2008-2013). Setelah menyelesaikan studi S1 penulis bekerja dibidang Data Cleaning dan Data Anlysis (Tahun 2013-2016). Kemudian penulis melanjutkan pendidikan ke jenjang S2 di Jurusan Statistika FMKSD Institut Teknologi Sepuluh Nopember (ITS) Surabaya (Agustus 2016-2018).